



SECP2523: DATABASE SEM I 2024/2025

DATABASE DESIGN DESCRIPTIONS: LOGICAL DATABASE DESIGN FOR KOPERASI KAKITANGAN KADA (KKK) ONLINE SYSTEM

Group Name: Tech Hi Five

Members:

- 1. Chua Jia Lin (Group Leader)**
- 2. Goe Jie Ying**
- 3. Lam Yoke Yu**
- 4. Tan Yi Ya**
- 5. Teh Ru Qian**

**Stakeholder: Encik Ahmad, Koperasi Kakitangan KADA
(KKK) Sdn Bhd**

Date: 27.1.2025

TABLE OF CONTENTS

1 Introduction	1
1.1 Overview about the company	1
1.2 The Existing System	1
1.3 The Proposed System	3
2 Global Conceptual Data Model (Global ERD)	7
2.1 Global Conceptual Database Design (Conceptual ERD)	7
2.2 Data Dictionary for the Global Conceptual ERD	8
3 Logical Data Model (Logical ERD)	14
3.1 Logical Database Design (Logical ERD)	15
3.2 Data Dictionary for the Logical ERD	15
3.3 Relational Database Schemas	26
3.4 Normalisation	28
4.0 Interface Design and SQL Implementation	35
4.1 Interface Design (Mapped to the Proposed SQL Statement)	35
Use Case: Log In	35
Use Case: Apply for membership (registration)	37
Use Case: View Loan Application Status	44
Use Case: Apply Loan	45
Use Case: View Financial Status	64
Use Case: Update User Profile	67
Use Case: Approve Membership	78
Use Case: Approve Loan Application	82
Use Case: Manage System	86
Use Case: Generate Report	98
4.2 SQL Statement	104
4 Summary	143
5 Appendices	144
5.1 Use Case Diagram	144
5.2 Activity Diagram	145
5.3 Meeting Log	150

1 Introduction

This document outlines the database design for the “KKK Online System”.

1.1 Overview about the company

Koperasi Kakitangan Daerah Kelantan Berhad (KKK) is a cooperative in Malaysia established to provide financial and social services to the employees of the Lembaga Kemajuan Pertanian Kemubu (KADA) and its members. The cooperative plays an important role in enhancing the economic well-being of its members through services such as savings, loan facilities and share offerings, exemplifying the principles of member ownership, democratic control and mutual benefit.

1.2 The Existing System

Currently, the cooperative uses a manual system for key processes such as new member registration, loan applications, and applications approval. Applicants must fill in physical member application forms and loan application forms, whereas the cooperative staff manually verify and check the eligibility of the information provided. After the meeting of the Board of Directors, the cooperative staffs inform the applicants about their application result. Figure 1, Figure 2, and Figure 3 are some of the physical forms and documents currently in use by the cooperative.

Figure 1: Member Application Form

KOPERASI KANTIGAN KADA KELANTAN BERHAD		
PERMOHONAN MENIADI ANGGOTA		
MAXIMUM 10 PERSOHONAN		
NAMA <input type="text"/> NO. KP <input type="text"/> TARAF PERKHAWIAN <input type="text"/> ALAMAT RUMAH <input type="text"/> POSKOD <input type="text"/> NEGERI <input type="text"/> NO. ANGGOTA <input type="text"/> NO. PF <input type="text"/> JAWATAN & GRED <input type="text"/> ALAMAT PEJABAT (Tempat Bertugas) <input type="text"/> POSKOD <input type="text"/> NO. TEL/FAX <input type="text"/> NO. TEL BIMBIT <input type="text"/> NO. TEL RUMAH <input type="text"/> GAJI BULANAN RM <input type="text"/>		
SILA BACA KETULOGKA DAN SETUJU		
* SILA BUAT LAMPIRAN JIKA RUJANGAN YANG DISEDIHKAN TIDAK MENCUCUP		
TERIMA SAMA		
JAKA DI TERIMA SEBAKI ANGGOTA, SAYA BERSETUJU MEMBAYAR YURAN DAN SUMBANGAN BULANAN SEPERTI DI BAWAH :		
BIL	PERKARA	RM
1	FEE MASUK	<input type="text"/>
2	MODAH SYER *	<input type="text"/>
3	MODAL YURAN	<input type="text"/>
4	WANG DEPOSIT ANGGOTA	<input type="text"/>
5	SUMBANGAN TABUNG KEBAJIKAN (AL-ABRAR)	<input type="text"/>
6	SIMPANAN TETAP	<input type="text"/>
7	LAIN-LAIN	<input type="text"/>
* MINIMA MODAL SYER ADALAH SEBANYAK RM 200.00 DAN TAHU MELAKU 1/2 DARAHADA MODAL SYER KOPERASI DAN HENDAKLAH DILELAUAN DALAM PEMERINTAHAN DAN DI LATAR KELUSSUN APENIANGGOTI		

Figure 2: Member Financing Application Form

	KOPERASI KAKTANGAN KADA KELANTAN BERHAD BORANG XXXX 02/00	BUTIR-BUTIR PEMBIAYAAN
PERMOHONAN PEMBIAYAAN ANGGOTA		
BUTIR-BUTIR PEMBIAYAAN		
BUTIR-BUTIR PEMBIAYAAN		
BUTIR-BUTIR PERBAIDI PEMOHON		
PENGESAHAN MAJAHAN		
LEGBALITI PEMOHON		
KETUGASAN JAWATAN		

Figure 3: Financial Statement Verification Form

 PENGESAHAN PENYATA KEWANGAN AHLI KOPERASI KAKITANGAN KADA KELANTAN BERHAD BAGI TAHUN BERAKHIR 30 JUN 2023.	NAMA: MOHD ROSLAN B ISMAIL NO. K/P: 740807-03-5479 NO. PF: 2273 NO. AHLI: 1579
Tuan/Puan, Untuk penentuan Juraudit, kami dengan ini menyatakan bagi akaun tuan/puan adalah sebagai berikut: MAKLUMAT SAHAM AHLI: Modah Syer: Modal Yuran: Simpanan Tetap: Tabung Anggota: Simpanan Anggota: RM RM RM RM RM .. MAKLUMAT PINJAMAN AHLI: Al-Bai: Al-Innah: B/Pulih Kenderaan: Road Tax& Insuran: Khas: Al-Qaduril Hassan: RM0.00 RM0.00 RM0.00 RM0.00 RM0.00 RM0.00	
PENGESAHAN BAGI PENYATA KEWANGAN AHLI KOPERASI KAKITANGAN KADA KELANTAN BERHAD BAGI TAHUN BERAKHIR 30 JUN 2023 Saya — MOHD ROSLAN B ISMAIL No. Ahli: 1579 mengesahkan bahawa Penyata Kewangan Koperasi Kakitangan KADA Kelantan Berhad bagi tahun berakhir 30 Jun 2023 adalah benar: Tandatangan: <input type="checkbox"/> Setuju Tarikh: <input type="checkbox"/> Tidak Setuju Nota: 1. Sila tandakan (/) di salah satu kotak diatas. 2. Sekiranya pihak Koperasi tidak menerima sebarang maklumbalas daripada tuan/puan sehingga 30 Jun 2023. maka pengesahan penyata ini dianggap betul dan tuan/puan bersetu.	

The manual system faced challenges such as the low work efficiency, high possibility of manual errors, information backup difficulties, and potential issues with data integrity. Therefore, the cooperative requires an automated system to address these problems.

1.3 The Proposed System

The proposed system, namely “KKK Online System” will replace the manual processes with a digital platform and transform key operations like member registration, loan applications, and financial reporting.

Mission Statement

The purpose of the KKK Online System is to manage the data that is used and generated to support the cooperative operation for our client and cooperative members.

Mission Objectives

To maintain (enter, update, and delete) data on members.

To maintain (enter, update, and delete) data on loans.

To maintain (enter, update, and delete) data on financial statuses.

To maintain (enter, update, and delete) data on policies.

To maintain (enter, update, and delete) data on banners.

To perform searches on members.

To perform searches on loans.

To track the status of members.

To track the status of loans.

To track the status of financial statuses.

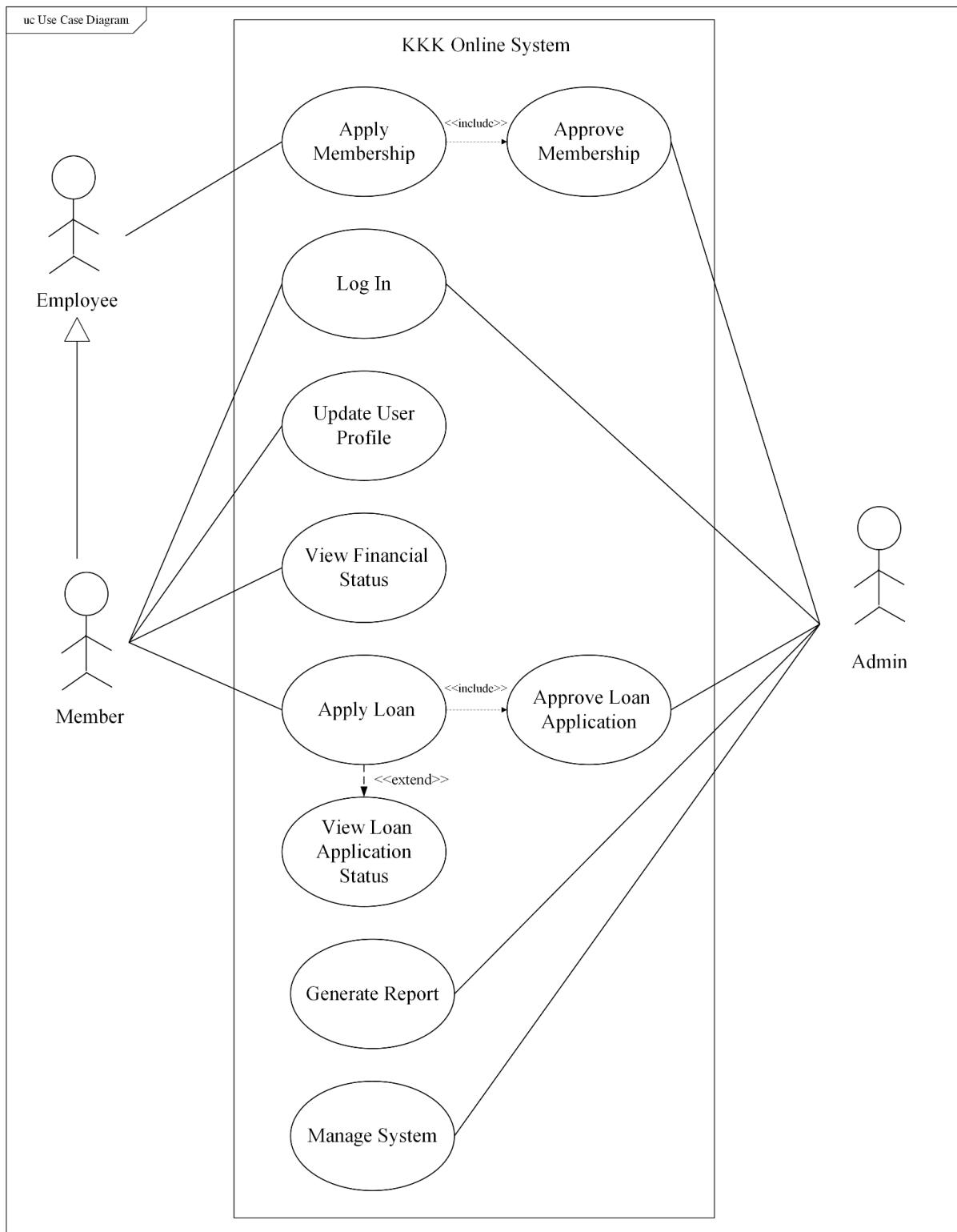
To report on members.

To report on loans.

To report on financial statuses.

System Boundaries

Figure 4: Use Case Diagram for KKK Online System



1. Apply Membership

The employees in KADA can apply to be a member of the KKK by filling in the online form and upload any required documents. Once their application is approved, the user would be able to log in to the system.

2. Login

The cooperative member and the administrator can log in the system by using the correct login credentials. The members will then be directed to the member's page and the administrator will be directed to the administrator's page.

3. Apply Loan

The cooperative members can apply for financing services from the cooperative by filling in the online form and upload any required documents. Upon submission, they would be able to see their application status through the system.

4. Update User Profile

The cooperative members can update their user personal information, and the family and inheritor information.

5. View Financial Status

Upon log in to the system, the cooperative member would be able to view their financial status including the share information (Maklumat Saham Ahli) and the loan information (Maklumat Pinjaman Ahli).

6. View List

The administrator would be able to view the list of membership and loan applications as well as the existing members and borrowers. Besides, the admin would be able to update the status of the membership and loan application.

7. Generate Report

The admin can generate monthly and annual reports that summarise the financial and membership information.

8. Manage System Record Transaction

This use case includes various administrative tasks, including recording transactions, updating policies, and updating banners.

After a transaction settled between the stakeholders, the admin would update the transaction in the system to reflect the transaction in the member profile. This includes updating the amount of share capital, funds, contributions and loans.

The administrator would be able to update policies such as the minimum membership, minimum share capital requirement for loan application, loan interest rate, maximum loan repayment duration and maximum financing amount.

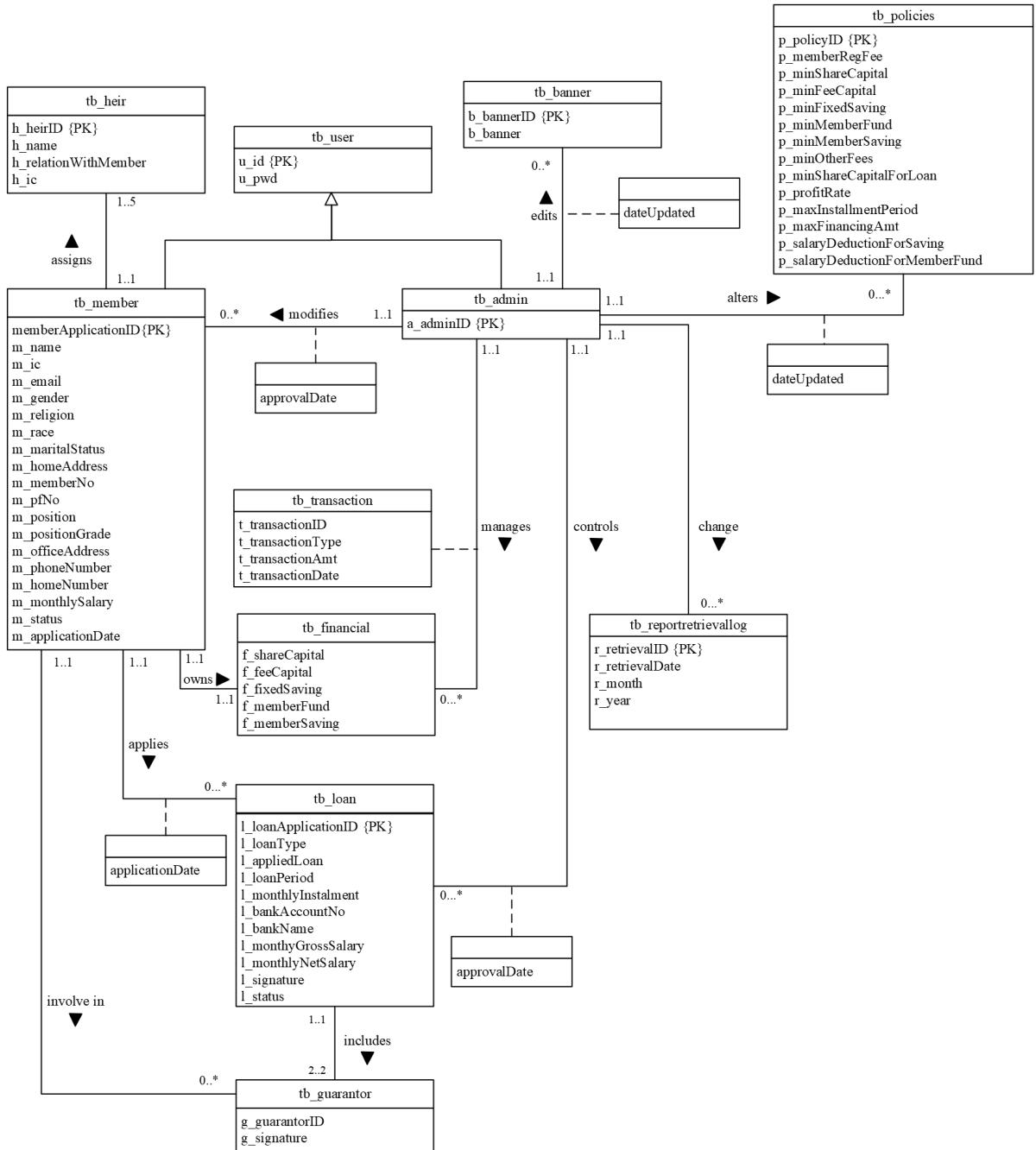
The administrator can update the website banner. This ensures the sustainability of the system.

Proposed Technology

For designing and developing the system, Figma is used to design the prototype, while technologies such as HTML, CSS, JavaScript, PHP and MySQL database are used to develop the system.

2 Global Conceptual Data Model (Global ERD)

2.1 Global Conceptual Database Design (Conceptual ERD)



2.2 Data Dictionary for the Global Conceptual ERD

2.2.1 Identifying Entity

Entity Name	Description	Aliases	Occurrence
tb_user	General term describing all users using KKK Online System	-	Each user can be defined by an admin or a member.
tb_member	General term describing all user logged in as member using KKK Online System	KKK Member	Each member assigns many heirs. Each member owns one financial record. Each member can apply for many loans. Each member can be a guarantor for many members. Each member application will be reviewed by an admin. Member is a type of user.
tb_admin	General term describing all user logged in as admin using KKK Online System	KKK Staff	Each admin edits many banners. Each admin alters many policies. Each admin retrieves many reports. Each admin can review and approve many loan and member applications. Each admin manages many financial records. Admin is a type of user.
tb_heir	General term describing all people declared as heir of a member applicant	-	Each heir is assigned by a member.
tb_banner	General term describing all banners or picture to be published on Member Main Page	Advertisement, Picture	Each banner will be updated by a single admin only.
tb_policies	General term describing all policies of the cooperative	-	Each policy will be altered by a single admin only.
tb_transaction	General term describing all transaction made by member	-	Each transaction record will be generated when a financial record exists.
tb_financial	General term describing	-	Each financial record will be

	all financial status of a member		managed by a single admin only.
tb_loan	General term describing all loans of the cooperative	-	Each loan application is reviewed by one admin.
tb_reportretrievallog	General term describing all report retrieval log of the system	-	Each report is retrieved by one admin.
tb_guarantor	General term describing all guarantors for a loan applicant	-	Each guarantor is included in one loan application. Each guarantor must be a member of KKK.

2.2.2 Identifying Relationship Types

Entity Name	Multiplicity	Relationship	Entity Name	Multiplicity
tb_member	1..1 1..1 1..1 1..1	assigns involve in applies owns	tb_heir tb_guarantor tb_loan tb_financial	1..5 0..* 0..* 1..1
tb_admin	1..1 1..1 1..1 1..1 1..1 1..1	edits alters change controls manages modifies	tb_banner tb_policies tb_reportretrievallog tb_loan tb_financial tb_member	0..* 0..* 0..* 0..* 0..* 0..*
tb_loan	1..1	includes	tb_guarantor	2..2

2.2.3 Attributes Description

Entity Name	Attributes	Data Type & Length	Description	Nulls	Multi-valued	Example
tb_user	u_id (PK)	int	Uniquely identifies each user	No	No	1
	u_pwd	varchar(255)	User password for login	No	No	Abc@123
tb_member	member Application	int	Uniquely identifies each member	No	No	1001

	onID (PK)		application		
m_name	varchar(50)	User's full name	No	No	Ali bin Abu
m_ic	varchar(14)	User's identification number	No	No	850101-08-1033
m_email	varchar(255)	User's email account	No	No	ali@gmail.com
m_gender	varchar(10)	User's gender	No	No	Lelaki
m_religion	varchar(10)	User's religion	No	No	Islam
m_race	varchar(10)	User's race	No	No	Melayu
m_maritalStatus	varchar(20)	User's marital status	No	No	Kahwin
m_homeAddress	varchar(255)	User's home address	No	No	1, Taman Bahagia, Jalan Bahagia, 57000 Kuala Lumpur
m_memberNo	int	Uniquely identifies each member	No	No	1
m_pfNo	int	User's PF number	No	No	1001
m_position	varchar(50)	User's position	No	No	Pengurus
m_positionGrade	varchar(11)	User's position grade	No	No	G41
m_officeAddress	varchar(255)	User's office address	No	No	25, Jalan Makmur, 57000 Kuala Lumpur
m_phone	varchar(13)	User's phone number	No	No	01234567

	Number					89
	m_home Number	varchar(13)	User's home number	Yes	No	051234567
	m_mont hlySalar y	double	User's monthly salary	No	No	3500
	m_status	varchar(20)	User's application status	No	No	Sedang Diproses
	m_applicationDat e	datetime	The date when user submit his/her member application	No	No	2025-01-03 16:00:00
tb_admin	a_admin ID (PK)	int	Uniquely identifies each admin	No	No	12345678
tb_heir	h_heirID (PK)	int	Uniquely identifies each heir	No	No	1
	h_name	varchar(50)	Heir's full name	No	No	Benjamin Tan
	h_relatio nWithM ember	varchar(20)	Heir's relation with member applicant	No	No	Anak
	h_ic	varchar(14)	Heir's identification number	No	No	000101-10-1233
tb_banner	b_banner ID (PK)	int	Uniquely identifies each banner	No	No	1
	b_banner	varchar(50)	Banner's picture	No	No	advertisement.jpg
tb_policies	p_policy ID (PK)	int	Uniquely identifies each policy	No	No	1
	p_membe rRegFee	double	Policy for member registration fee	No	No	35
	p_minSh areCapital	double	Policy for minimum share capital of member applicant	No	No	300

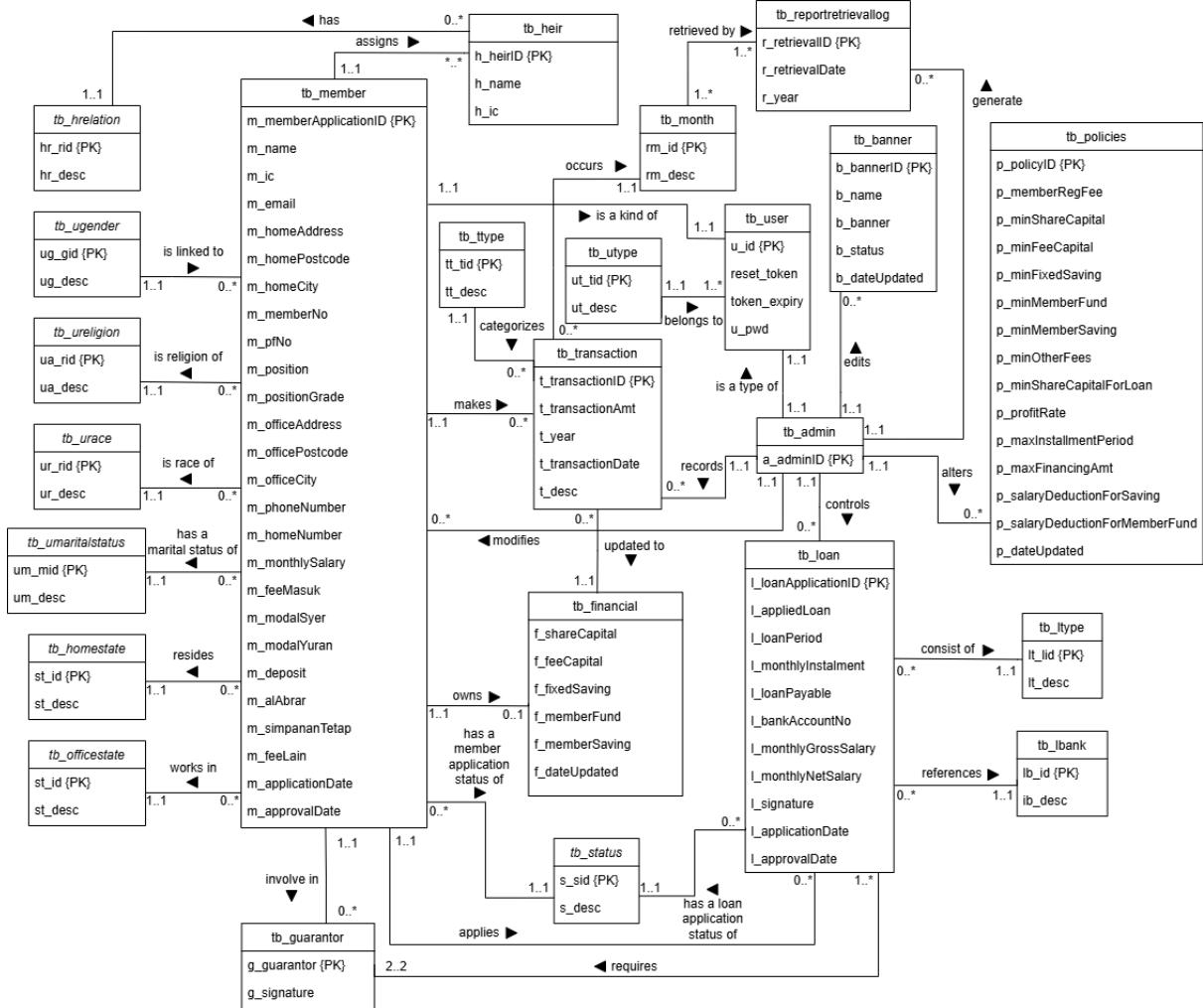
	p_minFeeCapital	double	Policy for minimum fee capital of member applicant	No	No	0
	p_minFixedSaving	double	Policy for minimum fixed saving of member applicant	No	No	0
	p_minMemberFund	double	Policy for minimum member fund of member applicant	No	No	5
	p_minMemberSaving	double	Policy for minimum member saving of member applicant	No	No	50
	p_minOtherFees	double	Policy for minimum other fees of member applicant	No	No	0
	p_minShareCapitalForLoan	double	Policy for minimum share capital loan of loan applicant	No	No	300
	p_profitRate	double	Policy for profit rate of loan	No	No	5
	p_maxInstalmentPeriod	int	Policy for maximum instalment period of loan applicant	No	No	6
	p_maxFinancingAmt	double	Policy for maximum financing amount of loan applicant	No	No	40000
	p_salaryDeductionForSaving	double	Policy for salary deduction for saving of loan borrower	No	No	50
	p_salaryDeductionForMemberFund	double	Policy for salary deduction for member fund of loan borrower	No	No	5
tb_transaction	t_transactionID	int	Uniquely identifies each transaction	No	No	1

	(PK)					
	t_transactionType	varchar(20)	Transaction type	No	No	Simpanan Tetap
	t_transactionAmt	double	Transaction amount	No	No	100
	t_transactionDate	timestamp	Transaction date	No	No	2024-12-20 17:30:00
tb_financial	f_shareCapital	double	Member's share capital	No	No	10
	f_feeCapital	double	Member's fee capital	No	No	100
	f_fixedSaving	double	Member's fixed saving	No	No	200
	f_memberFund	double	Member's fund	No	No	50
	f_memberSaving	double	Member's saving	No	No	200
tb_loan	l_loanApplication ID (PK)	int	Uniquely identifies each loan application	No	No	1
	l_loanType	varchar(20)	Loan type	No	No	al-Bai
	l_appliedLoan	double	Amount of loan applied	No	No	10000
	l_loanPeriod	int	Loan period	No	No	4
	l_monthlyInstalment	double	Monthly instalment for loan	No	No	90.33
	l_bankAccountNo	int	Loan borrower's bank account number	No	No	214748364
	l_bankName	varchar(20)	Loan borrower's bank	No	No	Affin Bank

	l_monthlyGrossSalary	double	Loan borrower's monthly gross salary	No	No	10000
	l_monthlyNetSalary	double	Loan borrower's monthly net salary	No	No	8000
	l_signature	varchar(50)	Loan borrower's signature	No	No	./uploads/8d969f4e9e3586a1226cc389a6a29b1b.jpeg
	l_status	varchar(20)	Loan application status	No	No	Sedang Diproses
tb_reportretrievallog	r_retrievalID (PK)	int	Uniquely identifies each retrieval record	No	No	1
	r_retrievalDate	date	Report retrieval date	No	No	2024-12-05
	r_month	varchar(10)	Selected month for report	No	No	November
	r_year	int	Selected year for report	No	No	2024
tb_guarantor	g_guarantorID (PK)	int	Uniquely identifies each guarantor	No	No	1
	g_signature	varchar(50)	Guarantor's signature	No	No	./uploads/678a24a16eee9.jpeg

3 Logical Data Model (Logical ERD)

3.1 Logical Database Design (Logical ERD)



3.2 Data Dictionary for the Logical ERD

3.2.1 Identifying Entity

Entity Name	Description	Aliases	Occurrence
tb_user	General term describing all users using KKK Online System	-	Admin and member are types of user. Each user has one user type.
tb_utype	General term describing all types of users using KKK Online System	User type	Each user type belongs to many users.

tb_member	General term describing all user logged in as member using KKK Online System	KKK Member, Member Application	Each member has one gender, one religion, one race and one marital status only. Each member resides and works in one state. Each member can become guarantors for other members. Each member applies for many loans. Each member application will have one status only at any one time. Each member owns at most one financial record. Each member application is approved by one admin. Each member can make many transactions. Members are one kind of user for the system. Each member can assign many heirs.
tb_ugender	General term describing the gender of a member	Gender	Each gender can be shared by many members.
tb_ureligion	General term describing the religion of a member	Religion	Each religion can be shared by many members.
tb_urace	General term describing the race of a member	Race	Each race can be shared by many members.
tb_umaritalstatus	General term describing the marital status of a member	Marital status	Each marital status can be shared by many members.
tb_homestate	General term describing the home state of a member	State	There can be none or many members residing in the state.
tb_officestate	General term describing the office state of a member	State	There can be none or many members working in the state.

tb_admin	General term describing all user logged in as admin using KKK Online System	KKK Staff	Admin is one type of user. Admin can edit many banners. Each admin can generate many reports. Each admin alters many policies. Each admin controls many loan applications. Each admin approves many member applications. Each admin can record many transaction records.
tb_heir	General term describing all people declared as heir of a member applicant	-	Each heir is assigned to one member. Each heir for a single member will have a relationship with members only.
tb_hrelation	General term describing the relationship between a member applicant and his heir	Relationship	Each heir relation with a member may be having by many members.
tb_banner	General term describing all banners or picture to be published on Member Main Page	-	Each banner is edited by one admin.
tb_policies	General term describing all policies of the cooperative	-	Each policy is altered by a single admin only.
tb_transaction	General term describing all transaction made by member	-	Each transaction has one transaction type only. Each transaction record occurs in one month. Each transaction record will be recorded by a single admin only. Each transaction made by one member. Each transaction record will be updated to a member's financial record only.
tb_ttype	General term describing all types of transactions	Transaction type	Each transaction type belongs to many transactions.
tb_financial	General term describing all financial status of a member	Financial status	Each financial record is owned by one member only. Each financial record can be updated by many transaction records.

tb_loan	General term describing all loans of the cooperative	Loan Application	Each loan is controlled by one admin. Each loan consists of one loan type. Each loan references a single bank only. Each loan application has a status at one time. Each loan application is applied by one member. Each loan application required two guarantors.
tb_ltype	General term describing all types of loans of the cooperative	Loan Type	Each loan type is being referenced by one or more loan applications.
tb_lbank	General term describing all banks	Bank	Each bank is being referenced by one or more loan applications.
tb_reportretrievallog	General term describing all report retrieval log of the system	Report	Each report will be retrieved by a single admin only. Each report may contain one or more month's summaries.
tb_guarantor	General term describing all guarantors for a loan applicant	-	Two guarantors are required for each loan application. Each guarantor is involved by one member only.
tb_status	General term describing all status for the applications	-	Each membership and loan application has a single status at any one time.
tb_month	General term describing all month in a year	-	Each month can be retrieved by one or more reports. Each month may be contained in one or more transaction records.

3.2.2 Identifying Relationship Types

Entity Name	Multiplicity	Relationship	Entity Name	Multiplicity
tb_member	1..* 1..1 1..1 1..1 1..1 0..*	assigns involve in applies owns makes links	tb_heir tb_guarantor tb_loan tb_financial tb_transaction tb_ugender	*..* 0..* 0..* 0..1 0..* 1..1

	0..*	is religion of	tb_ureligion	1..1
	0..*	is race of	tb_urace	1..1
	0..*	has a marital status of	tb_umaritalstatus	1..1
	0..*	resides	tb_homestate	1..1
	0..*	works in	tb_officestate	1..1
	0..*	has a member application status of	tb_status	1..1
	1..1	is a kind of	tb_user	1..1
tb_admin	1..1	edits	tb_banner	0..*
	1..1	alters	tb_policies	0..*
	1..1	generate	tb_reportretrievallog	0..*
	1..1	controls	tb_loan	0..*
	1..1	record	tb_financial	0..*
	1..1	modifies	tb_member	0..*
	1..1	is a type of	tb_user	1..1
tb_user	1..1	is having	tb_utype	1..*
tb_heir	0..*	has	tb_hrelation	1..1
tb_loan	1..1	includes	tb_guarantor	2..2
	0..*	consists of	tb_ltype	1..1
	0..*	references	tb_lbank	1..1
	0..*	has a loan status of	tb_status	1..1
tb_transaction	0..*	categorizes	tb_ttype	1..1
	0..*	updated to	tb_financial	1..1
	0..*	contains	tb_month	1..1

3.2.3 Attributes Description

Entity Name	Attributes	Data Type & Length	Description	Nulls	Multi-valued	Example
tb_user	u_id (PK)	int (11)	Uniquely identifies each user	No	No	1
	reset_token	varchar (255)	Used when user are going to reset the password	Yes	No	1440f2ccd0064f390b491ca7cae30b5
	token_expi	timestamp	Record the time	Yes	No	2025-11-1

	ry		when user reset password			1 16:00:00
	u_pwd	varchar(255)	User password for login	No	No	Abc@123
tb_admin	a_adminID (PK)	int (11)	Uniquely identifies each admin	No	No	200
tb_utype	ut_tid (PK)	int (11)	Uniquely identifies each user type	No	No	1
	ut_desc	varchar (10)	User's type description	No	No	Admin
tb_member	memberApplicationID (PK)	int (11)	Uniquely identifies each member application	No	No	1001
	m_name	varchar(50)	User's full name	No	No	Ali bin Abu
	m_ic	varchar(14)	User's identification number	No	No	850101-08-1033
	m_homeAddress	varchar(255)	User's home address	No	No	1, Taman Bahagia, Jalan Bahagia,
	m_homePostcode	int (11)	User's home postcode	No	No	57000
	m_homeCity	varchar (50)	User's home city	No	No	Kuala Lumpur
	m_memberNo	int	Uniquely identifies each member	No	No	1
	m_pfNo	int	User's PF number	No	No	1001
	m_position	varchar(50)	User's position	No	No	Pengurus
	m_positionGrade	varchar(11)	User's position grade	No	No	G41
	m_officeA	varchar(255)	User's office	No	No	25, Jalan

	ddress		address			Makmur
	m_officePostcode	int (11)	User's office postcode	No	No	57000
	m_officeCity	varchar (50)	User's office city	No	No	Kuala Lumpur
	m_phoneNumber	varchar(13)	User's phone number	No	No	0123456789
	m_homeNumber	varchar(13)	User's home number	Yes	No	051234567
	m_monthlySalary	double	User's monthly salary	No	No	3500.00
	m_feeMasuk	double	User's entry fee	No	No	45.00
	m_modalSyer	double	User's shares	No	No	300.00
	m_modalYuran	double	User's share fee	No	No	50.00
	m_deposit	double	User's deposit	No	No	50.00
	m_alAbrar	double	A kind of user's fund	No	No	5.00
	m_simpanTetap	double	User's fixed saving	No	No	0.00
	m_feeLain	doube	User's other fee	No	No	0.00
	m_applicationDate	datetime	The date when user submit his/her member application	No	No	2025-01-03 16:00:00
	m_approvalDate	timestamp	The date when the member application being approved by admin	Yes	No	2025-01-15 06:40:00
tb_ugender	ug_gid (PK)	int (11)	The unique id for gender	No	No	1
	ug_desc	varchar (10)	User's gender	No	No	Lelaki

			description			
tb_ureligion	ua_rid (PK)	int (11)	The unique id for religion	No	No	1
	ua_desc	varchar (10)	User's religion description	No	No	Islam
tb_urace	ur_rid (PK)	int (11)	The unique id for race	No	No	1
	ur_desc	varchar (10)	User gender's description	No	No	Lelaki
tb_umaritalstatus	um_mid (PK)	int (11)	The unique id for marital status	No	No	1
	um_desc	varchar (10)	User's marital status description	No	No	Bujang
tb_homestate	st_id (PK)	int (11)	The unique id for user's home state	No	No	1
	st_desc	varchar (15)	User's home state description	No	No	Johor
tb_officestate	st_id (PK)	int (11)	The unique id for user's office state	No	No	1
	st_desc	varchar (15)	User's office state description	No	No	Melaka
tb_heir	h_heirID (PK)	int (11)	Uniquely identifies each heir	No	No	1
	h_name	varchar(50)	Heir's full name	No	No	Tan Mei Ling
	h_ic	varchar(14)	Heir's identification number	No	No	010101-01-0101
tb_hrelation	hr_rid (PK)	int (11)	Uniquely identifies each heir relationship with member	No	No	1
	hr_desc	varchar (20)	The description of heir relationship with member	No	No	Suami Isteri

tb_banner	b_bannerID (PK)	int	Uniquely identifies each banner	No	No	1
	b_name	varchar (255)	Banner's name	No	No	Sunny 16 boat
	b_banner	varchar(50)	Banner's picture	No	No	advertisement.jpg
	b_status	tinyint (1)	Banner's active or inactive state	No	No	0
	b_dateUpdated	timestamp	Banner's updated time	No	No	2025-01-19 11:31:43
tb_policies	p_policyID (PK)	int	Uniquely identifies each policy	No	No	1
	p_memberRegFee	double	Policy for member registration fee	No	No	35
	p_minShareCapital	double	Policy for minimum share capital of member applicant	No	No	300
	p_minFeeCapital	double	Policy for minimum fee capital of member applicant	No	No	0
	p_minFixedSaving	double	Policy for minimum fixed saving of member applicant	No	No	0
	p_minMemberFund	double	Policy for minimum member fund of member applicant	No	No	5
	p_minMemberSaving	double	Policy for minimum member saving of member applicant	No	No	50
	p_minOtherFees	double	Policy for minimum other fees of member applicant	No	No	0
	p_minShareCapitalForLoan	double	Policy for minimum share capital loan of loan applicant	No	No	300

	p_profitRate	double	Policy for profit rate of loan	No	No	5
	p_maxInstallmentPeriod	int	Policy for maximum instalment period of loan applicant	No	No	6
	p_maxFinancingAmt	double	Policy for maximum financing amount of loan applicant	No	No	40000
	p_salaryDeductionForSaving	double	Policy for salary deduction for saving of loan borrower	No	No	50
	p_salaryDeductionForMemberFund	double	Policy for salary deduction for member fund of loan borrower	No	No	5
	p_dateUpdated	timestamp	The date when the admin update the policy	No	No	2024-12-20 17:30:00
tb_transaction	t_transactionID (PK)	int	Uniquely identifies each transaction	No	No	1
	t_transactionAmt	double	Transaction amount	No	No	100
	t_year	int (11)	The year when the transaction made	No	No	2024
	t_transactionDate	timestamp	Transaction date	No	No	2024-12-20 17:30:00
	t_desc	varchar (255)	The objective of the transaction made	NO		Potongan Gaji Bayaran Balik 8
tb_ttype	tt_lid (PK)	int	Uniquely identifies each transaction type	No	No	1
	tt_desc	varchar (30)	The description of	No	No	Al-Qadrul

			each transaction type			Hassan
tb_financial	f_shareCapital	double	Member's share capital	No	No	10
	f_feeCapital	double	Member's fee capital	No	No	100
	f_fixedSaving	double	Member's fixed saving	No	No	200
	f_memberFund	double	Member's fund	No	No	50
	f_memberSaving	double	Member's saving	No	No	200
	f_dateUpdated	double	The date when the financial record is updated	No	No	2025-01-21 15:36:25
tb_loan	l_loanApplicationID (PK)	int	Uniquely identifies each loan application	No	No	1
	l_appliedLoan	double	Amount of loan applied	No	No	10000
	l_loanPeriod	int	Loan period	No	No	4
	l_monthlyInstalment	double	Monthly instalment for loan	No	No	90.33
	l_loanPutable	double	The total loan that need to be paid by the member	No	No	1897.01
	l_bankAccountNo	int	Loan borrower's bank account number	No	No	214748364
	l_monthlyGrossSalary	double	Loan borrower's monthly gross salary	No	No	10000
	l_monthlyNetSalary	double	Loan borrower's monthly net salary	No	No	8000

	l_signature	varchar(50)	Loan borrower's signature	No	No	./uploads/8d969f4e9e3586a1226cc389a6a29b1b.jpeg
	l_file	varchar (50)	Loan borrower's employer's verification proof	No	No	pengesahan_majikan_4_1737462659.pdf
	l_applicationDate	datetime	The date where loan application is made by a member	No	No	2025-01-05 12:51:36
	l_approvalDate	timestamp	The date where loan application is approved by admin	Yes	No	2025-01-11 23:40:25
tb_ltype	lt_lid (PK)	int	Uniquely identifies each retrieval record	No	No	1
	lt_desc	varchar (30)	The description of each loan type	No	No	Al-Qadrul Hassan
tb_lbanks	lb_lid (PK)	int	Uniquely identifies each bank	No	No	1
	lb_desc	varchar (30)	The description of each bank	No	No	BSN
tb_reportretrievallog	r_retrievalID (PK)	int (11)	Uniquely identifies each retrieval log	No	No	1
	r_retrievalDate	date	Report retrieval date	No	No	2024-12-05
	r_year	int	Selected year for report	No	No	2024
tb_guarantor	g_guarantorID (PK)	int (11)	Uniquely identifies each guarantor	No	No	1
	g_signature	varchar(50)	Guarantor's signature	No	No	./uploads/678a24a1

						6eee9.jpeg
tb_statuses	s_sid (PK)	int (11)	Uniquely identifies each status	No	No	2
	s_desc	varchar (20)	The description of each status	No	No	Ditolak
tb_rmonth	rm_id (PK)	int (11)	Uniquely identifies each month	No	No	3
	rm_desc	varchar (10)	The description of each month	No	No	Mac

3.3 Relational Database Schemas

1NF

tb_admin (a_adminID)

PK: a_adminID

tb_banner (b_bannerID, b_name, b_banner, b_status, b_dateUpdated, b_adminID)

PK: b_bannerID

FK: b_adminID references tb_admin (a_adminID)

tb_financial (f_memberNo, f_shareCapital, f_feeCapital, f_fixedSaving, f_memberFund, f_memberSaving, f_dateUpdated)

FK: f_memberNo references tb_member (m_memberNo)

tb_guarantor (g_guarantorID, g_loanApplicationID, g_memberNo, g_signature)

PK: g_guarantorID

FK: g_loanApplicationID references tb_loan (l_loanApplicationID)

FK: g_memberNo references tb_member (m_memberNo)

tb_heir (h_heirID, h_memberApplicationID, h_name, hr_rid, hr_desc, , h_ic)

PK: h_heirID

FK: h_memberApplicationID references tb_member (m_memberApplicationID)

tb_loan (l_loanApplicationID, l_memberNo, lt_lid, lt_desc, l_appliedLoan, l_loanPeriod, l_monthlyInstalment, l_loanPayable, l_bankAccountNo, lb_id, lb_desc, l_monthlyNetSalary, l_signature, l_file, s_sid, s_desc, l_applicationDate, l_approvalDate, l_adminID)

PK: l_loanApplicationID

FK: l_memberNo references tb_member (m_memberNo)

FK: l_adminID references tb_admin (a_adminID)

tb_member (m_memberApplicationID, m_name, m_ic, m_email, ug_gid, ug_desc, ua_rid, ua_desc, ur_rid, ur_desc, um_mid, um_desc, m_homeAddress, m_homePostcode, m_homeCity, st_id, st_desc, m_memberNo, m_pfNo, m_position, m_positionGrade,

m_officeAddress, m_officePostcode, m_officeCity, st_id, st_desc, m_phoneNumber,
m_homeNumber, m_monthlySalary, m_feeMasuk, m_modalSyer, m_modalYuran,
m_deposit, m_alAbrar, m_simpananTetap, m_feeLain, m_status, m_applicationDate,
m_approvalDate, m_adminID)

PK: m_memberApplicationID

FK: m_adminID references tb_admin (a_adminID)

tb_policies (p_policyID, p_memberRegFee, p_minShareCapital, p_minFeeCapital,
p_minFixedSaving, p_minMemberFund, p_minMemberSaving, p_minOtherFees,
p_minShareCapitalForLoan, p_profitRate, p_maxInstallmentPeriod, p_maxFinancingAmt,
p_salaryDeductionForSaving, p_salaryDeductionForMemberFund, p_dateUpdated,
p_adminID)

PK: p_policyID

FK: p_adminID references tb_admin (a_adminID)

tb_reportretrievelog (r_retrievalID, r_retrievalDate, rm_id, rm_desc, r_year, r_adminID)

PK: r_retrievalID

FK: r_adminID references tb_admin (a_adminID)

tb_transaction (t_transactionID, tt_id, tt_desc, t_transactionAmt, rm_id, rm_desc, t_year,
t_desc, t_transactionDate, t_memberNo, t_adminID)

PK: t_transactionID

FK: t_memberNo references tb_member (m_memberNo)

FK: t_adminID references tb_admin (a_adminID)

tb_user (u_id, reset_token, token_expiry, u_pwd, ut_tid, ut_desc)

PK: u_id

3.4 Normalisation

2NF

tb_admin (a_adminID)

PK: a_adminID

tb_banner (b_bannerID, b_name, b_banner, b_status, b_dateUpdated, b_adminID)

PK: b_bannerID

FK: b_adminID references tb_admin (a_adminID)

tb_financial (f_memberNo, f_shareCapital, f_feeCapital, f_fixedSaving, f_memberFund,
f_memberSaving, f_dateUpdated)

FK: f_memberNo references tb_member (m_memberNo)

tb_guarantor (g_guarantorID, g_loanApplicationID, g_memberNo, g_signature)

PK: g_guarantorID

FK: g_loanApplicationID references tb_loan (l_loanApplicationID)

FK: g_memberNo references tb_member (m_memberNo)

tb_heir (h_heirID, h_memberApplicationID, h_name, hr_rid, hr_desc, , h_ic)

PK: h_heirID

FK: h_memberApplicationID references tb_member (m_memberApplicationID)

tb_loan (l_loanApplicationID, l_memberNo, lt_lid, lt_desc, l_appliedLoan, l_loanPeriod, l_monthlyInstalment, l_loanPayable, l_bankAccountNo, lb_id, lb_desc, l_monthlyNetSalary, l_signature, l_file, s_sid, s_desc, l_applicationDate, l_approvalDate, l_adminID)

PK: l_loanApplicationID

FK: l_memberNo references tb_member (m_memberNo)

FK: l_adminID references tb_admin (a_adminID)

tb_member (m_memberApplicationID, m_name, m_ic, m_email, ug_gid, ug_desc, ua_rid, ua_desc, ur_rid, ur_desc, um_mid, um_desc, m_homeAddress, m_homePostcode, m_homeCity, st_id, st_desc, m_memberNo, m_pfNo, m_position, m_positionGrade, m_officeAddress, m_officePostcode, m_officeCity, st_id, st_desc, m_phoneNumber, m_homeNumber, m_monthlySalary, m_feeMasuk, m_modalSyer, m_modalYuran, m_deposit, m_alAbrar, m_simpananTetap, m_feeLain, m_status, m_applicationDate, m_approvalDate, m_adminID)

PK: m_memberApplicationID

FK: m_adminID references tb_admin (a_adminID)

tb_policies (p_policyID, p_memberRegFee, p_minShareCapital, p_minFeeCapital, p_minFixedSaving, p_minMemberFund, p_minMemberSaving, p_minOtherFees, p_minShareCapitalForLoan, p_profitRate, p_maxInstallmentPeriod, p_maxFinancingAmt, p_salaryDeductionForSaving, p_salaryDeductionForMemberFund, p_dateUpdated, p_adminID)

PK: p_policyID

FK: p_adminID references tb_admin (a_adminID)

tb_reportretrievelog (r_retrievalID, r_retrievalDate, rm_id, rm_desc, r_year, r_adminID)

PK: r_retrievalID

FK: r_adminID references tb_admin (a_adminID)

tb_transaction (t_transactionID, tt_id, tt_desc, t_transactionAmt, rm_id, rm_desc, t_year, t_desc, t_transactionDate, t_memberNo, t_adminID)

PK: t_transactionID

FK: t_memberNo references tb_member (m_memberNo)

FK: t_adminID references tb_admin (a_adminID)

tb_user (u_id, reset_token, token_expiry, u_pwd, ut_tid, ut_desc)

PK: u_id

3NF

tb_admin (a_adminID)

PK: a_adminID

tb_banner (b_bannerID, b_banner, b_dateUpdated, b_adminID)

PK: b_bannerID

FK: b_adminID references tb_admin(a_adminID)

tb_financial (f_memberNo, f_shareCapital, f_feeCapital, f_fixedSaving, f_memberFund, f_memberSaving, f_alBai, f_allnnah, f_bPulihKenderaan, f_roadTaxInsurance, f_specialScheme, f_specialSeasonCarnival, f_alQadrulHassan, f_dateUpdated)

PK: f_memberNo

FK: f_memberNo references tb_member(m_memberNo)

tb_guarantor (g_guarantorID, g_loanApplicationID, g_memberNo, g_signature)

PK: g_guarantorID

FK: g_loanApplicationID references tb_loan(l_loanApplicationID)

FK: g_memberNo references tb_member(m_memberNo)

tb_heir (h_heirID, h_memberApplicationID, h_name, h_relationWithMember, h_ic)

PK: h_heirID

FK: h_memberApplicationID references tb_member (m_memberApplicationID)

FK: h_relationWithMember references tb_hrelation (hr_rid)

tb_hrelation (hr_rid, hr_desc)

PK: hr_rid

tb_loan (l_loanApplicationID, l_memberNo, l_loanType, l_appliedLoan, l_loanPeriod, l_monthlyInstalment, l_loanPayable, l_bankAccountNo, l_bankName, l_monthlyNetSalary, l_signature, l_file, l_status, l_applicationDate, l_approvalDate, l_adminID)

PK: l_loanApplicationID

FK: l_loanType references tb_ltype (lt_lid)

FK: l_status references tb_status (s_sid)

FK: l_memberNo references tb_member (m_memberNo)

FK: l_bankName references tb_lbank (lb_id)

FK: l_adminID references tb_admin (a_adminID)

tb_ltype (lt_lid, lt_desc)

PK: lt_lid

tb_lbank (lb_id, lb_desc)

PK: lb_id

tb_member (m_memberApplicationID, m_name, m_ic, m_email, m_gender, m_religion, m_race, m_maritalStatus, m_homeAddress, m_homePostcode, m_homeCity, m_homeState, m_memberNo, m_pfNo, m_position, m_positionGrade, m_officeAddress, m_officePostcode, m_officeCity, m_officeState, m_phoneNumber, m_homeNumber, m_monthlySalary, m_feeMasuk, m_modalSyer, m_modalYuran, m_deposit, m_alAbrar, m_simpananTetap, m_feeLain, m_status, m_applicationDate, m_approvalDate, m_adminID)

PK: m_memberApplicationID

FK: m_gender references tb_ugender (ug_gid)

FK: m_religion references tb_ureligion (ua_rid)

FK: m_race references tb_urace (ur_rid)

FK: m_maritalStatus references tb_umaritalstatus (um_mid)

FK: m_status references tb_status (s_sid)

FK: m_homeState references tb_homestate (st_id)

FK: m_officeState references tb_officestatus (st_id)

FK: m_adminID references tb_admin (a_adminID)

tb_ugender (ug_gid, ug_desc)

PK: ug_gid

tb_umaritalstatus (um_mid, um_desc)

PK: um_mid

tb_urace (ur_rid, ur_desc)

PK: ur_rid

tb_ureligion (ua_rid, ua_desc)

PK: ua_rid

tb_homestate (st_id, st_desc)

PK: st_id

tb_officestate (st_id, st_desc)

PK: st_id

tb_status (s_sid, s_desc)

PK: s_sid

tb_policies (p_policyID, p_memberRegFee, p_minFixedSaving, p_minMemberFund, p_minMemberSaving, p_minOtherFees, p_minShareCapitalForLoan, p_profitRate, p_maximumInstallmentPeriod, p_maxFinancingAmt, p_salaryDeductionForSaving, p_salaryDeductionForMemberFund, p_dateUpdated, p_adminID)

PK: p_policyID

FK: p_adminID references tb_admin(a_adminID)

tb_reportretrievallog (r_retrievalID, r_retrievalDate, r_month, r_year, r_adminID)

PK: r_retrievalID

FK: r_adminID references tb_admin(a_adminID)

FK: month references tb_rmonth(month)

tb_rmonth (rm_id, rm_desc)

PK: rm_id

tb_transaction (t_transactionID, t_transactionType, t_transactionAmt, t_transactionDate, t_memberNo, t_adminID)

PK: t_transactionID

FK: t_memberNo references tb_member(m_memberNo)

FK: t_adminID references tb_admin(a_adminID)

tb_ttype (tt_id, tt_desc)

PK: tt_id

tb_user (u_id, reset_token, token_expiry, u_pwd, u_type)

PK: u_id

FK: u_type references tb_utype(ut_tid)

tb_utype (ut_tid, ut_desc)

PK: ut_tid

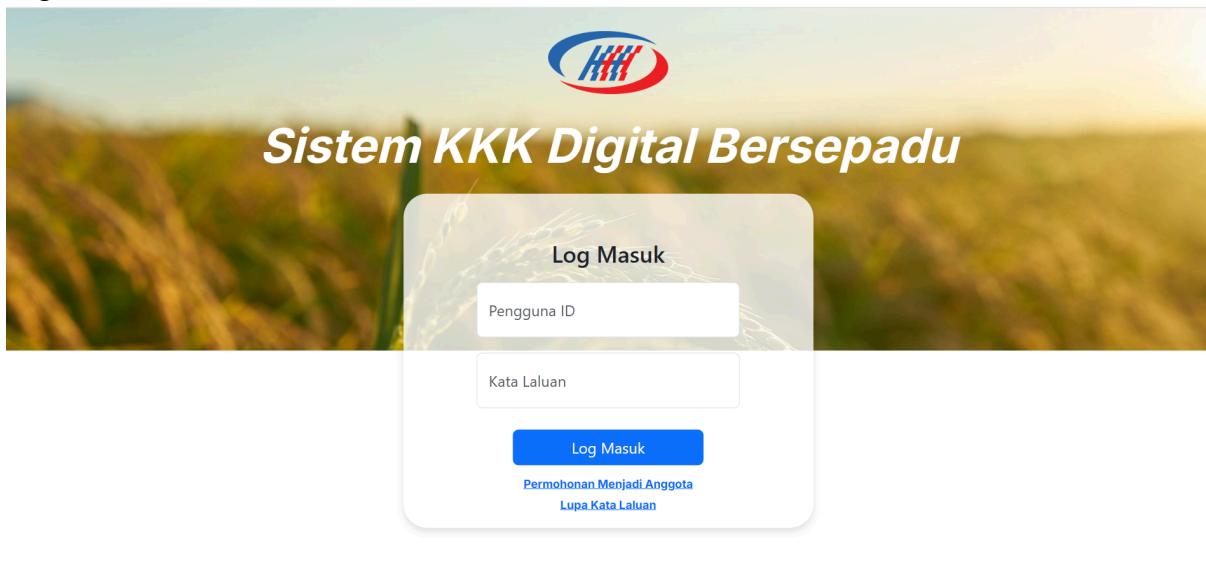
4.0 Interface Design and SQL Implementation

4.1 Interface Design (Mapped to the Proposed SQL Statement)

4.1.1 Member View

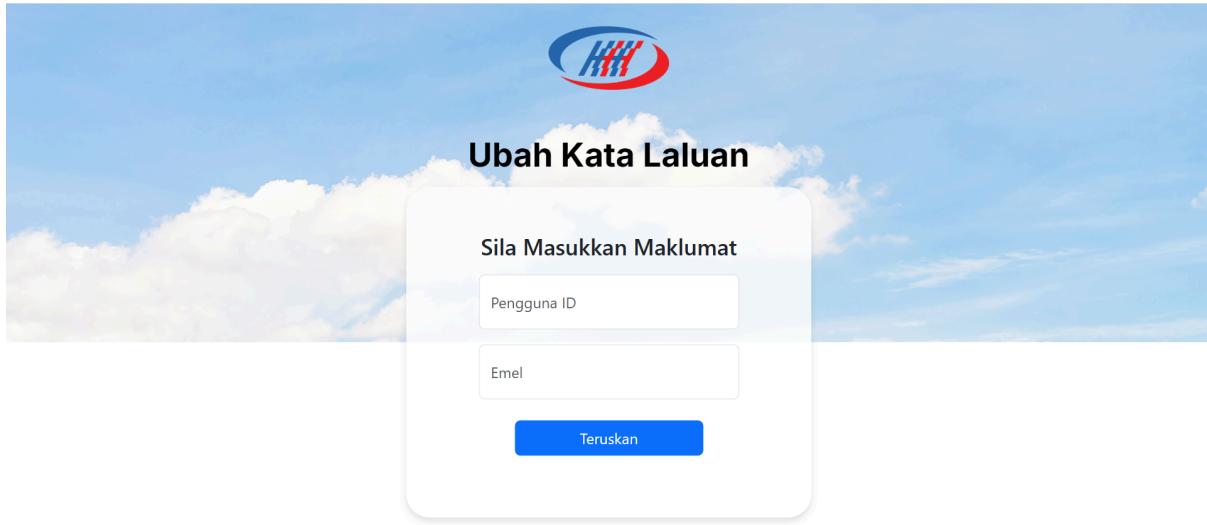
Use Case: Log In

Log in Dashboard



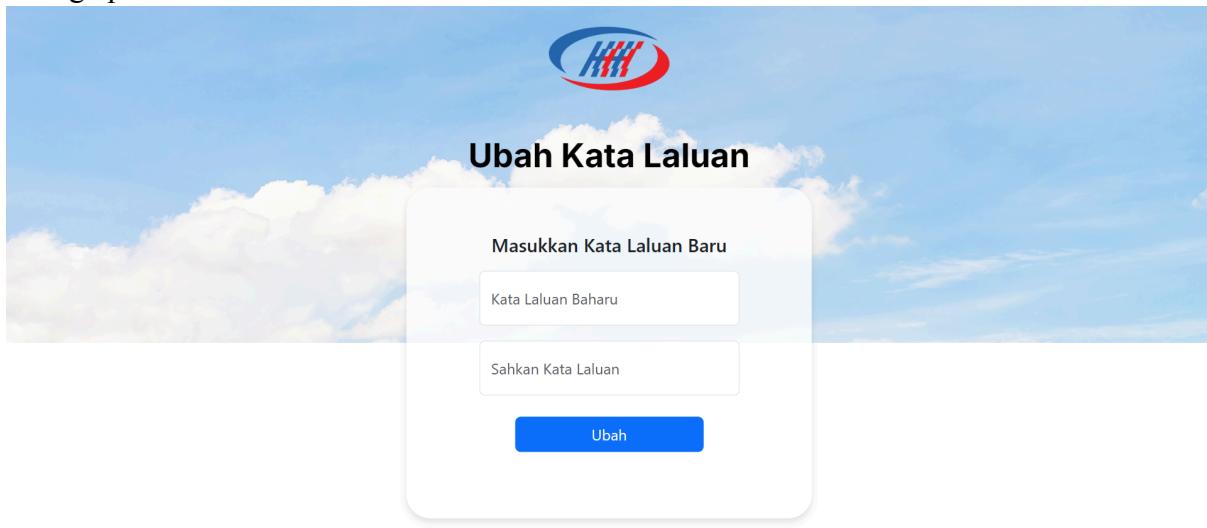
```
1 //Retrieves data from tb_user for user availability by user id
2
3 $sql = "SELECT * FROM tb_user WHERE u_id = ?";
4 $stmt = mysqli_prepare($con, $sql);
5 mysqli_stmt_bind_param($stmt, 'i', $funame);
6 mysqli_stmt_execute($stmt);
7 $result = mysqli_stmt_get_result($stmt);
8 $user = mysqli_fetch_assoc($result);
9 $matches = false;
10
11 if($user){
12     $matches = password_verify($fpwd, $user['u_pwd']);
```

Forgot password Dashboard



```
1 //Retrieves data from tb_member for matching memberId and email
2
3 $sql = "SELECT m_memberNo, m_email
4     FROM tb_member
5     WHERE m_memberNo = ? AND m_email = ?";
6     $stmt = mysqli_prepare($con, $sql);
7     mysqli_stmt_bind_param($stmt, 'is', $userID, $email);
8     mysqli_stmt_execute($stmt);
9     $result = mysqli_stmt_get_result($stmt);
10
11    if (mysqli_num_rows($result) > 0) {
12        //generate a unique token that last 10 min
13        $token = bin2hex(random_bytes(16));
14        $expiryTime = date("Y-m-d H:i:s", strtotime("+10 minutes",
15 time()));
16
17
18 //Set reset_token and expiry time in tb_user by userID
19
20         $sql = "UPDATE tb_user SET reset_token = ?, token_expiry = ?
21 WHERE u_id = ?";
22         $stmt = mysqli_prepare($con, $sql);
23         mysqli_stmt_bind_param($stmt, 'ssi', $token, $expiryTime,
24 $userID);
25         mysqli_stmt_execute($stmt);
```

Change password dashboard



```
1 //check for matching token and expiry status in tb_user by reset token
2
3 $sql = "SELECT * FROM tb_user WHERE reset_token = ?";
4 $stmt = mysqli_prepare($con, $sql);
5 mysqli_stmt_bind_param($stmt, 's', $token);
6 mysqli_stmt_execute($stmt);
7 $result = mysqli_stmt_get_result($stmt);
8 $user = mysqli_fetch_assoc($result);
9
10 //reset password by reset token
11 $sql = "UPDATE tb_user SET u_pwd = ?, reset_token = NULL,
12 token_expiry = NULL WHERE reset_token = ?";
13 $stmt = mysqli_prepare($con, $sql);
14 mysqli_stmt_bind_param($stmt, 'ss', $hashedPassword, $token);
15 mysqli_stmt_execute($stmt);
16
17
```

Use Case: Apply for membership (registration)

Registration Interface

The screenshots show a registration form with a sidebar menu on the left.

Screenshot 1: Personal Information (Maklumat Pemohon)

- Maklumat Pemohon
- Maklumat Pewaris
- Yuran dan Sumbangan
- Akaun Kebenaran

Maklumat Pemohon

Nama: Seperti dalam IC

No. Kad Pengenalan: XXXXX-XX-XXXX

Emel:

Jantina:

Lelaki Perempuan

Screenshot 2: Religious and Nationality Information (Agama, Bangsa, Taraf Perkahwinan)

- Maklumat Pemohon
- Maklumat Pewaris
- Yuran dan Sumbangan
- Akaun Kebenaran

Agama:

Islam Buddha Hindu Kristian Lain-lain

Bangsa:

Melayu Cina India Lain-lain

Taraf Perkahwinan:

Alamat Rumah:

Bandar: [dropdown], Negeri: [dropdown], Poskod: [dropdown]

Screenshot 3: Employment and Address Details (Jawatan, Alamat Pejabat, Bandar, Negeri, Poskod, Telefon, Gaji)

- Maklumat Pemohon
- Maklumat Pewaris
- Yuran dan Sumbangan
- Akaun Kebenaran

Jawatan: [dropdown], Gred Jawatan: [dropdown], PF No.: [dropdown]

Alamat Pejabat:

Bandar: [dropdown], Negeri: [dropdown], Poskod: [dropdown]

No. Telefon: Nombor Telefon Tanpa '-' | No. Telefon Rumah: Talian Rumah Tanpa '-'

Gaji Sebulan: RM 0.00

Buttons at the bottom:

Simpan **Seterusnya**

```

1 //finds for similar ic and email in tb_member to check for duplicated
2 entries
3 $fuic = $cleanPost['fuic'];
4 $sql = "SELECT * FROM tb_member WHERE m_ic = ?";
5 $binds = [$fuic];
6 $result = query($sql, $binds);
7
8 $row = mysqli_fetch_assoc($result);
9
10 if ($row) {
11 //display error message
12 }
13
14 $femail = $cleanPost['femail'];
15 $sql_email = "SELECT * FROM tb_member WHERE m_email = ?";
16 $binds_email = [$femail];
17 $result_email = query($sql_email, $binds_email);
18
19 if ($row_email = mysqli_fetch_assoc($result_email)) {
20     display error message
21 }
22
23

```

Registration: Fees Interface

The screenshot displays a user interface for fee and donation registration. The main title is "Yuran dan Sumbangan". The form contains several input fields with validation notes:

- Fee Masuk:** RM 35.00 (Nilai Minima: 35.00)
- Modal Syer:** RM 300.00
- Modal Yuran:** RM 0.00 (Nilai Minima: 0.00)
- Wang Deposit Anggota:** RM 50.00 (Nilai Minima: 50.00)
- Sumbangan Tabung Kebajikan (Al-Abrar):** RM 5.00 (Nilai Minima: 5.00)

The sidebar on the left provides navigation links:

- Maklumat Pemohon
- Maklumat Pewaris
- Yuran dan Sumbangan** (highlighted in blue)
- Akaun Kebenaran

```

1 //retrieves a single latest minimum value of fees from tb_policies
2
3 $minValues = [];
4 $query = "
5     SELECT
6         p_memberRegFee AS ffee,
7         p_minShareCapital AS fmodal,
8         p_minFeeCapital AS fyuran,
9         p_minFixedSaving AS fanggota,
10        p_minMemberFund AS fabrar,
11        p_minMemberSaving AS ftetap,
12        p_minOtherFees AS fother
13    FROM tb_policies
14    ORDER BY p_policyID DESC
15    LIMIT 1";
16
17 $result = mysqli_query($con, $query);
18 if ($result && $row = mysqli_fetch_assoc($result)) {
19     $minValues = $row;
20 }

```

SQL implementation of Registration

```

1 include('..\db_connect.php');
2     $fdate = date('Y-m-d H:i:s');
3
4     // Start transaction
5     $con->begin_transaction();
6
7     // Insert into tb_member
8     $stmt = $con->prepare("
9         INSERT INTO tb_member (
10             m_name, m_ic, m_email, m_gender, m_religion, m_race,
11             m_maritalStatus, m_homeAddress, m_homeCity, m_homeState,
12             m_homePostcode,
13             m_position, m_positionGrade, m_pfNo, m_officeAddress,
14             m_officeCity, m_officeState, m_officePostcode,

```

```

15      m_phoneNumber, m_homeNumber, m_monthlySalary, m_feeMasuk,
16      m_modalSyer, m_modalYuran, m_deposit,
17      m_alAbrar, m_simpananTetap, m_feeLain, m_applicationDate,
18      m_status
19      ) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?,
20      ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)
21      ");
22
23      $stmt->bind_param(
24          "ssssssssssssssssssssssssssssssi",
25          $funame,
26          $fuic,
27          $femail,
28          $fgender,
29          $freligion,
30          $frace,
31          $fmariage,
32          $fhomeaddress,
33          $fcity,
34          $fstate,
35          $fhomezip,
36          $fposition,
37          $fgrade,
38          $fpfno,
39          $fofficeaddress,
40          $fofficecity,
41          $fofficestate,
42          $fofficezip,
43          $ftelno,
44          $fhomeno,
45          $fsalary,
46          $ffee,
47          $fmodal,
48          $fyuran,
49          $fanggota,
50          $fabrar,
51          $ftetap,
52          $fother,
53          $fdate,
54          $fstatus
55      );
56
57      if ($stmt->execute()) {
58          // Get the last inserted member ID
59          $member_id = $con->insert_id;
60
61          // Prepare heir insert statement
62          $stmt_heir = $con->prepare("
63              INSERT INTO tb_heir (h_memberApplicationID, h_name, h_ic,
64              h_relationWithMember)
65              VALUES (?, ?, ?, ?, ?)
66          ");
67
68          if ($stmt_heir) {
69              // Insert heirs
70              foreach ($heirs as $heir) {
71

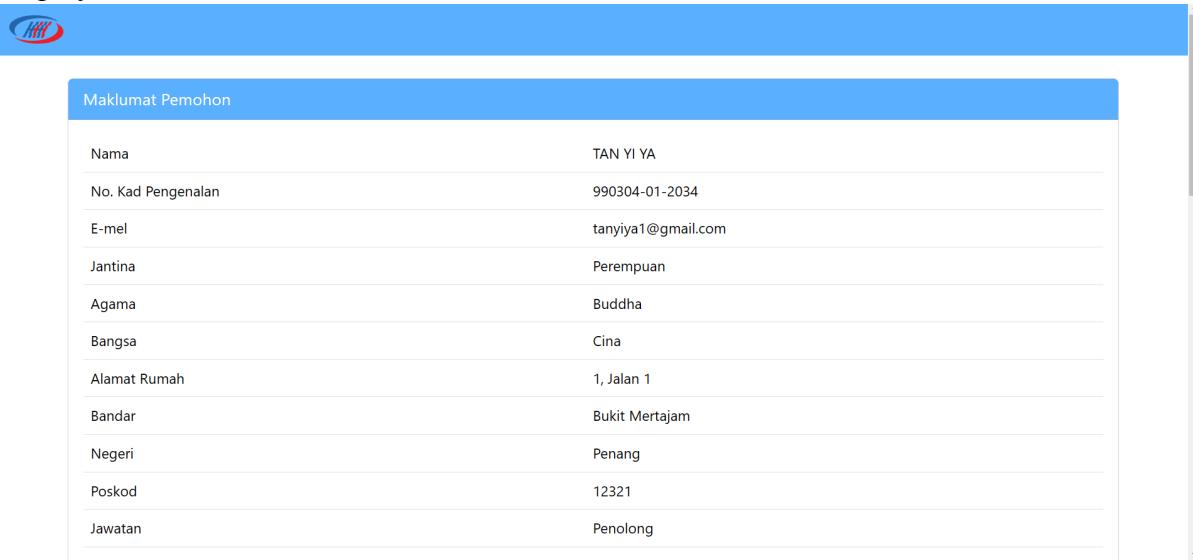
```

```

72             $stmt_heir->bind_param("isss", $member_id,
73 $heir['name'], $heir['ic'], $heir['relation']);
74             if (!$stmt_heir->execute()) {
75                 // Rollback on error
76                 $con->rollback();
77                 echo "<script>alert('Error inserting heir data: " .
78 $stmt_heir->error . "'');</script>";
79                 exit();
80             }
81         }
82
83         // Commit the transaction
84         $con->commit();

```

Display Details Interface



The screenshot shows a web page titled 'Display Details Interface'. At the top left is a logo consisting of three red and blue overlapping shapes. The main content area has a light gray background. A blue header bar contains the title. Below it is a table with a light gray border and white background, titled 'Maklumat Pemohon' (Applicant Information). The table lists the following details:

Maklumat Pemohon	
Nama	TAN YI YA
No. Kad Pengenalan	990304-01-2034
E-mel	tanyiya1@gmail.com
Jantina	Perempuan
Agama	Buddha
Bangsa	Cina
Alamat Rumah	1, Jalan 1
Bandar	Bukit Mertajam
Negeri	Penang
Poskod	12321
Jawatan	Penolong

Gaji Sebulan	RM 3000
--------------	---------

Yuran dan Sumbangan	
Yuran Masuk	RM 35
Modal Syer	RM 300
Yuran	RM 0
Anggota	RM 50
Al Abrar	RM 5
Simpanan Tetap	RM 50
Lain-lain	RM 0

Maklumat Waris		
Nama	No. Kad Pengenalan	Hubungan
TAN MAMA	991202-01-8372	Orang Tua
Anggota		RM 50
Al Abrar		RM 5
Simpanan Tetap		RM 50
Lain-lain		RM 0

Maklumat Waris		
Nama	No. Kad Pengenalan	Hubungan
TAN MAMA	991202-01-8372	Orang Tua
TAN MI MI	024739-59-3720	Orang Tua
CHOUCHOU	247392-73-9483	Anak

[Selesai](#)

```

1 if (isset($_GET['member_id']) && !empty($_GET['member_id'])) {
2     $member_id = intval($_GET['member_id']); // Sanitize input
3
4     // Retrieve member details from tb_member by memberID for display
5     purpose
6     $stmt = $con->prepare("SELECT * FROM tb_member
7         LEFT JOIN tb_ugender ON tb_member.m_gender = tb_ugender.ug_gid
8         LEFT JOIN tb_ureligion ON tb_member.m_religion = tb_ureligion.ua_rid
9         LEFT JOIN tb_urace ON tb_member.m_race = tb_urace.ur_rid
10        LEFT JOIN tb_homestate ON tb_member.m_homeState = tb_homestate.st_id
11        LEFT JOIN tb_officestate ON tb_member.m_officeState =
12            tb_officestate.st_id
13        WHERE m_memberApplicationID = ?");
14        $stmt->bind_param("i", $member_id);
15        $stmt->execute();
16        $member_result = $stmt->get_result();
17        $member_data = $member_result->fetch_assoc();
18

```

```

19 // Retrieve heir details from tb_heir by memberID for display purpose
20
21 $stmt = $con->prepare("SELECT * FROM tb_heir
22 LEFT JOIN tb_hrelation ON tb_heir.h_relationWithMember =
23 tb_hrelation.hr_rid
24 WHERE h_memberApplicationID = ?");
25 $stmt->bind_param("i", $member_id);
26 $stmt->execute();
27 $heirs_result = $stmt->get_result();
28 $heirs_data = $heirs_result->fetch_all(MYSQLI_ASSOC);
29
30 $stmt->close();
31 }
32
33
34

```

Use Case: View Loan Application Status

Loan Dashboard

The screenshot shows a web application interface for loan management. At the top, there is a navigation bar with links for Laman Utama, Profil, Pinjaman, and Log Keluar. Below the navigation is a table titled "Papan Pemuka Pinjaman" (Pending Loan Table) showing four rows of loan applications:

No.	Jenis Pinjaman	Jumlah Permohonan (RM)	Tempoh Pinjaman (Bulan)	Ansuran Bulanan (RM)	Tunggakan (RM)	Tarikh Permohonan	Status
1	Baik Pulih Kenderaan	15000	2	677.5	16260	2025-12-03 00:00:00	Sedang Diproses
2	Al-Bai	50	2	5	55	2025-01-01 00:00:00	Dilulus
3	Al-Bai	50	2	5	55	2025-01-02 00:00:00	Dilulus
4	Baik Pulih Kenderaan	2222	4	55.55	2666.4	2025-01-19 12:34:17	Ditolak

At the bottom left is a blue button labeled "Mohon Pinjaman". The footer is blue and contains three sections: "ALAMAT" with address details, "HUBUNGI KAMI" with contact info (phone 09-1234567, email info@kada.gov.my), and "WAKTU OPERASI" with operating hours (Monday-Friday 08:00-15:30, Saturday & Sunday Closed). Social media icons for Facebook, YouTube, and Instagram are also present.

```

1 //Retrieve data from tb_loan for dashboard loan
2 $sql = "SELECT l_loanType, l_appliedLoan, l_loanPeriod,
3 l_monthlyInstalment, l_loanPayable, l_status, l_applicationDate
4     FROM tb_loan
5     WHERE l_memberNo= $memberNo";
6 $result = mysqli_query($con, $sql);
7
8 //Retrieve data from tb_ltype for loan type
9 if ($selected_jenis_pembayaran_ID) {
10 $sql_desc = "SELECT lt_desc
11             FROM tb_ltype
12             WHERE lt_lid = $selected_jenis_pembayaran_ID";
13 $desc_result = mysqli_query($con, $sql_desc);
14

```

```

15 }
16 //Retrieve data from tb_status for loan status
17 if ($status_loan_ID) {
18     $sql_status = "SELECT s_desc
19                     FROM tb_status
20                     WHERE s_sid = $status_loan_ID";
21     $status_result = mysqli_query($con, $sql_status);
22 }
23

```

Use Case: Apply Loan

Loan details

The screenshots show a user interface for applying a loan. The top screenshot shows initial input fields: 'Jenis Pembiayaan' (Al-Bai), 'Amaun Dipohon*' (RM 0.00), 'Tempoh Pembiayaan*' (0 tahun), and 'Ansuran Bulanan*' (RM 0.00). The bottom screenshot shows additional fields: 'Nama Bank/Cawangan' (Affin Bank), 'Bank Account' (00000000), 'Gaji Kasar Bulanan' (RM 0.00), 'Gaji Bersih Bulanan' (RM 0.00), and a 'Tandatangan' section with a file upload field ('Choose File') showing 'No file chosen'. Both screenshots include a sidebar with navigation links like 'Butir-Butir Pembiayaan', 'Butir-Butir Peribadi Pemohon', etc., and a header with 'Laman Utama', 'Profil', 'Pinjaman', and 'Log Keluar'.

Laman Utama Profil Pinjaman Log Keluar

Butir-Butir Pembiayaan

Butir-Butir Peribadi Pemohon

Pengakuan Pemohon

Butir-Butir Penjamin

Pengesahan Majikan

Akuan Kebenaran

RM 0.00

Gaji Bersih Bulanan

RM 0.00

Tandatangan

Choose File No file chosen

*Fail yang dibenarkan adalah dalam format PNG, JPG dan JPEG sahaja. Sila pastikan saiz fail tidak melebihi 5MB.

Simpan

ALAMAT
Peti Surat 127,
Bandar Kota Bharu
15710 Kota Bharu
Kelantan

HUBUNGI KAMI
Telefon: 09-1234567
Email: info@kada.gov.my

IKUTI KAMI

WAKTU OPERASI
Ahad - Rabu: 08:00 - 17:00
Khamis: 08:00 - 15:30
Jumaat: Rehat
Sabtu: Rehat

Hak Cipta © 2025 KADA. Semua Hak Terpelihara.

```

1 // Retrieve data from tb_policies and tb_financial to make sure
2 // applicant able to apply loan and check for loan details does not //
3 exceed in tb_policies
4
5 $sql = "SELECT * FROM tb_policies
6     ORDER BY p_policyID DESC
7     LIMIT 1;";
8 $result = mysqli_query($con, $sql);
9 $policy = mysqli_fetch_assoc($result);
10
11 $sql = "SELECT f_shareCapital
12     FROM tb_financial
13     WHERE f_memberNo = $memberNo";
14 $result = mysqli_query($con, $sql);
15
16 // Insert data to tb_loan
17 $sql = "INSERT INTO tb_loan (l_memberNo, l_loanType, l_appliedLoan,
18 l_loanPeriod, l_monthlyInstalment, l_loanPayable, l_bankAccountNo,
19 l_bankName, l_monthlyGrossSalary, l_monthlyNetSalary, l_signature,
20 l_status, l_applicationDate)
21     VALUES ('$memberNo', '$jenis_pembiayaan', '$amaunDipohon',
22 '$tempohPembiayaan', '$ansuranBulanan', '$tunggakan', '$bankAcc',
23 '$namaBank', '$gajiKasar', '$gajiBersih', '$fileSign', 1,
CURRENT_TIMESTAMP())";

```

Check and update personal details

The screenshots show a user profile page with a sidebar menu and three different sections for updating personal information.

Screenshot 1: Personal Information

- Left Sidebar:** Butir-Butir Pembayaan, Butir-Butir Peribadi Pemohon, Pengakuan Pemohon, Butir-Butir Penjamin, Pengesahan Majikan, Akuan Kebenaran.
- Section Header:** **Butir-Butir Peribadi Pemohon**
- Fields:**
 - Nama: Ravi Kumar
 - No. Kad Pengenalan: 870401-06-0555
 - Email: ravikumar@gmail.com
 - Jantina: Lelaki Perempuan
 - Agama: Islam Buddha Hindu Kristian Lain-lain
 - Bangsa: Melayu Cina India Lain-lain
 - Taraf Perkahwinan: Bujang Kahwin Cerai Kematian Pasangan
 - No. Anggota: 3

Screenshot 2: Address Information

- Left Sidebar:** Butir-Butir Pembayaan, Butir-Butir Peribadi Pemohon, Pengakuan Pemohon, Butir-Butir Penjamin, Pengesahan Majikan, Akuan Kebenaran.
- Fields:**
 - No. PF: 1005
 - Alamat Rumah: No.25, Taman Desa Aman
 - Poskod: 54000
 - Bandar: Kuala Lumpur
 - Negeri: WP Kuala Lumpur
 - Jawatan Gred: J41

The application interface consists of two main sections:

- Top Section (Form Input):**
 - Left sidebar menu: Butir-Butir Pembayaan, Butir-Butir Peribadi Pemohon, Pengakuan Pemohon, Butir-Butir Penjamin, Pengesahan Majikan, Akuan Kebenaran.
 - Form fields:
 - Jawatan: Pegawai Teknikal
 - Alamat Pejabat (Tempat Bertugas): Jalan Dato Lundang
 - Poskod: 15000
 - Bandar: Kota Bharu
 - Negeri: WP Kuala Lumpur
 - No. Telefon Rumah: 0182221234
- Bottom Section (Summary):**
 - Left sidebar menu: Butir-Butir Pembayaan, Butir-Butir Peribadi Pemohon, Pengakuan Pemohon, Butir-Butir Penjamin, Pengesahan Majikan, Akuan Kebenaran.
 - Form fields:
 - No. Telefon Bimbit: 0182221234
 - No. Telefon Bimbit: 0163344556
 - Gaji Bulanan: RM 4500
 - Buttons: Simpan
 - Footer sections:
 - ALAMAT:** Peti Surat 127, Bandar Kota Bharu 15710 Kota Bharu Kelantan
 - HUBUNGI KAMI:** Telefon: 09-1234567, Email: info@kada.gov.my
 - IKUTI KAMI:** Facebook, Instagram, YouTube icons
 - WAKTU OPERASI:** Ahad - Rabu: 08:00 - 17:00, Kamis: 08:00 - 15:30, Jumaat: Rehat, Sabtu: Rehat
 - Page footer: Hak Cipta © 2025 KADA. Semua Hak Terpelihara.

```

1 //Retrieve data from tb_member for checking details
2 if ($memberNo !== null) {
3     $sql = "SELECT m_name, m_ic, m_email, m_gender, m_religion, m_race,
4     m_maritalStatus, m_pfNo, m_monthlySalary, m_homeAddress,
5     m_homePostcode, m_homeCity, m_homeState, m_positionGrade, m_position,
6     m_officeAddress, m_officePostcode, m_officeCity, m_officeState,
7     m_homeNumber, m_phoneNumber
8     FROM tb_member
9     WHERE m_memberNo = '$memberNo'";
10 $result = mysqli_query($con, $sql);
11
12 // Modify data in tb_member
13 if (!empty($memberNo)) {
14     $sql = "UPDATE tb_member
15         SET m_name = '$nama',
16             m_email = '$email',
17             m_gender = '$selectedGender',
18             m_religion = '$selectedReligion',
19             m_race = '$selectedRace',

```

```

20      m_maritalStatus = '$selectedMaritalRace',
21      m_monthlySalary = '$selectedMonthlySalary',
22
23      m_homeAddress = '$selectedHomeAdd',
24      m_homePostcode = '$selectedHomePostcode',
25      m_homeCity = '$selectedHomeCity',
26      m_homeState = '$selectedHomeState',
27
28      m_position = '$selectedPosition',
29      m_positionGrade = '$selectedPositionGrade',
30
31      m_officeAddress = '$selectedOfficeAdd',
32      m_officePostcode = '$selectedOfficePostcode',
33      m_officeCity = '$selectedOfficeCity',
34      m_officeState = '$selectedOfficeState',
35
36      m_phoneNumber = '$selectedNoTeleBim',
37      m_homeNumber = '$selectedNoTelePej'
38 WHERE m_memberNo = '$memberNo'";
39
40 }

```

Applicant's statement

The screenshot shows a web-based application for loan applications. On the left, there is a vertical sidebar with navigation links: 'Butir-Butir Pembiayaan', 'Butir-Butir Peribadi Pemohon', 'Pengakuan Pemohon' (which is the active tab), 'Butir-Butir Penjamin', 'Pengesahan Majikan', and 'Akuan Kebenaran'. The main content area has a blue header bar with the title 'Pengakuan Pemohon'. Below the header, there are input fields for 'Saya' (Name) containing 'Ravi Kumar' and 'No. Kad Pengenalan' (NRIC Number) containing '870401-06-0555'. A large text area contains a declaration in Bahasa Melayu, which roughly translates to: 'I, Ravi Kumar, hereby give my consent to KOPERASI KAKITANGAN KADA KELANTAN BHD to act as my agent to receive any documents or notices related to this loan application. I also agree to be liable for any amount borrowed and to pay it back to the company. I have read and understood the terms and conditions of this loan application.' At the bottom of this text area is a radio button labeled 'Setuju' (Agree). Below the declaration text, there are three buttons: 'Simpan' (Save), 'ALAMAT' (Address), 'HUBUNGI KAMI' (Contact Us), and 'WAKTU OPERASI' (Operating Hours).

```

1 // Retrieve data from tb_member
2 if ($memberNo !== null) {
3     $sql = "SELECT m_name, m_ic
4         FROM tb_member
5         WHERE m_memberNo = '$memberNo'";
6     $result = mysqli_query($con, $sql);
7 }

```

Guarantor details

```

1 //Retrieve data from tb_member for taking guarantor 1 details by "No.
2 //anggota"
3 if(isset($_POST['anggotaPenjamin1'])) {$memberNo =
4 mysqli_real_escape_string($con, $_POST['anggotaPenjamin1']);
5
6     $sql = "SELECT m_memberNo, m_name, m_ic, m_pfNo
7         FROM tb_member
8         WHERE m_memberNo = ?";
9
10    $stmt = mysqli_prepare($con, $sql);
11    mysqli_stmt_bind_param($stmt, "s", $memberNo);
12    mysqli_stmt_execute($stmt);
13    $result = mysqli_stmt_get_result($stmt);
14
15    if(mysqli_num_rows($result) > 0) {
16        $member = mysqli_fetch_assoc($result);
17        echo json_encode([
18            'status' => 'success',
19            'penjamin1' => [

```

```

20             'm_name' => $member['m_name'],
21             'm_ic' => $member['m_ic'],
22             'm_pfNo' => $member['m_pfNo']
23         ]
24     ]);
25 } else {
26     echo json_encode([
27         'status' => 'error',
28         'message' => 'Tiada maklumat anggota dijumpai'
29     ]);
30 }
31 mysqli_stmt_close($stmt);
32 }

33
34 //Retrieve data from tb_member for taking guarantor 2 details by "No.
35 //anggota"
36 if(isset($_POST['anggotaPenjamin2'])) {$memberNo =
37 mysqli_real_escape_string($con, $_POST['anggotaPenjamin2']);
38
39     $sql = "SELECT m_memberNo, m_name, m_ic, m_pfNo
40             FROM tb_member
41             WHERE m_memberNo = ?";
42
43     $stmt = mysqli_prepare($con, $sql);
44     mysqli_stmt_bind_param($stmt, "s", $memberNo);
45     mysqli_stmt_execute($stmt);
46     $result = mysqli_stmt_get_result($stmt);
47
48     if(mysqli_num_rows($result) > 0) {
49         $member = mysqli_fetch_assoc($result);
50         echo json_encode([
51             'status' => 'success',
52             'penjamin2' => [
53                 'm_name' => $member['m_name'],
54                 'm_ic' => $member['m_ic'],
55                 'm_pfNo' => $member['m_pfNo']
56             ]
57         ]);
58     } else {
59         echo json_encode([
60             'status' => 'error',
61             'message' => 'Tiada maklumat anggota dijumpai'
62         ]);
63     }
64     mysqli_stmt_close($stmt);
65 }
66
67 //Insert data to tb_guarantor
68 $sql1 = "INSERT INTO tb_guarantor (g_loanApplicationID, g_memberNo,
69 g_signature)
70         VALUES ('$loanApplicationID', '$memberNo1',
71 '$fileSignPenjamin1')";
72
73 $sql2 = "INSERT INTO tb_guarantor (g_loanApplicationID, g_memberNo,
74 g_signature)
75         VALUES ('$loanApplicationID', '$memberNo2',
76 '$fileSignPenjamin2')";

```

Employer verification

The screenshot shows two identical web pages side-by-side, likely demonstrating a design change or a different state of the application.

Left Side (Original Design):

- Header:** Laman Utama, Profil, Pinjaman, Log Keluar.
- Left Sidebar:** Butir-Butir Pembiayaan, Butir-Butir Peribadi Pemohon, Pengakuan Pemohon, Butir-Butir Penjamin, Pengesahan Majikan, Akuan Kebenaran.
- Form Fields:**
 - Gaji Kasar Bulanan: RM 123
 - Gaji Bersih Bulanan: RM 456555
 - Muat Naik Pengesahan Majikan: Choose File (No file chosen)
 - Note: *Fail yang dibenarkan adalah dalam format PDF sahaja. Sila pastikan saiz fail tidak melebihi 5MB.
- Buttons:** Simpan.
- Contact Information:**
 - ALAMAT:** Peti Surat 127, Bandar Kota Bharu
 - HUBUNGI KAMI:** Telefon: 09-1234567, Email: info@kada.gov.my
 - WAKTU OPERASI:** Ahad - Rabu: 08:00 - 17:00, Kamis: 08:00 - 15:30

Right Side (Modified Design):

- Header:** Laman Utama, Profil, Pinjaman, Log Keluar.
- Left Sidebar:** Butir-Butir Pembiayaan, Butir-Butir Peribadi Pemohon, Pengakuan Pemohon, Butir-Butir Penjamin, Pengesahan Majikan, Akuan Kebenaran.
- Form Fields:**
 - Gaji Kasar Bulanan: RM 123
 - Gaji Bersih Bulanan: RM 456555
 - Muat Naik Pengesahan Majikan: Choose File (No file chosen)
 - Note: *Fail yang dibenarkan adalah dalam format PDF sahaja. Sila pastikan saiz fail tidak melebihi 5MB.
- Buttons:** Simpan.
- Contact Information:**
 - ALAMAT:** Peti Surat 127, Bandar Kota Bharu, 15710 Kota Bharu, Kelantan
 - HUBUNGI KAMI:** Telefon: 09-1234567, Email: info@kada.gov.my
 - IKUTI KAMI:** Facebook, Instagram, YouTube
 - WAKTU OPERASI:** Ahad - Rabu: 08:00 - 17:00, Kamis: 08:00 - 15:30, Jumaat: Rehat, Sabtu: Rehat

Hak Cipta © 2025 KADA. Semua Hak Terpelihara.

```

1 //Retrieve data from tb_loan for checking details
2 $sql = "SELECT l_monthlyGrossSalary, l_monthlyNetSalary
3     FROM tb_loan
4     WHERE l_loanApplicationID = $loanApplicationID";
5     $result = mysqli_query($con, $sql);

```

Statement of truth

Laman Utama Profil Pinjaman Log Keluar

Butir-Butir Pembayaran
Butir-Butir Peribadi Pemohon
Pengakuan Pemohon
Butir-Butir Penjamin
Pengesahan Majikan
Akuan Kebenaran

Akuan Kebenaran

Saya mengaku bahawa semua maklumat yang diberi adalah benar dan betul. Sekiranya saya didapati memberikan maklumat tidak benar atau palsu, saya boleh disabitkan kesalahan di bawah seksyen 193 Kanun Keseksianan (Akta 574) dan boleh dikenakan hukuman penjara selama tempoh yang boleh sampai tiga (3) tahun dan boleh juga dikenakan denda.

Setuju

Simpan

ALAMAT
Peti Surat 127,
Bandar Kota Bharu
15710 Kota Bharu
Kelantan

HUBUNGKAMI
Telefon: 09-1234567
Email: info@kada.gov.my

WAKTU OPERASI
Ahad – Rabu: 08:00 – 17:00
Khamis: 08:00 – 15:30
Jumaat: Rehat
Sabtu: Rehat

Hak Cipta © 2025 KADA. Semua Hak Terpelihara.

Details Check

Laman Utama Profil Pinjaman Log Keluar

Semakan Butiran

Butir-butir Pembayaran

Jenis Pembayaran	Al-Bai	Kemaskini
Amoun Dipohon	RM 1234	
Tempoh Pembayaran	2 bulan	
Ansuran Bulanan	RM 56.56	
Nama Bank/Cawangan	Hong Leong Bank	
Bank Account	90876589	
Gaji Kasar Bulanan	RM 123	
Gaji Bersih Bulanan	RM 456555	
Tandatangan		

Laman Utama Profil Pinjaman Log Keluar

Butir-butir Peribadi Pemohon

Nama	Ravi Kumar	Kemaskini
No. Kad Pengenalan	870401-06-0555	
Email	ravikumar@gmail.com	
Jantina	Lelaki	
Agama	Hindu	
Bangsa	India	
Taraf Perkahwinan	Kahwin	
No. Anggota	3	
No. PF	1005	
Alamat Rumah	No.25, Taman Desa Aman	
Poskod	54000	
Bandar	Kuala Lumpur	
Negeri	WP Kuala Lumpur	
Jawatan Gred	J41	

KADA

Laman Utama Profil Pinjaman Log Keluar

Negeri	WP Kuala Lumpur
Jawatan Gred	J41
Jawatan	Pegawai Teknikal
Alamat Pejabat (Tempat Bertugas)	Jalan Dato Lundang
Poskod	15000
Bandar	Kota Bharu
No. Telefon Rumah	0182221234
No. Telefon Bimbit	0163344556
Gaji Bulanan	RM 4500

Butir-butir Penjamin

Kemaskini

Penjamin 1	
No. Anggota	2
Nama	Mohd Hafiz
No. Kad Pengenalan	880701-05-0444

KADA

Laman Utama Profil Pinjaman Log Keluar

Nama	Mohd Hafiz
No. Kad Pengenalan	880701-05-0444
No. PF	1004
Tandatangan	
Penjamin 2	
No. Anggota	5
Nama	Lim Wei Sheng
No. Kad Pengenalan	850301-08-0777
No. PF	1007
Tandatangan	

KADA

Laman Utama Profil Pinjaman Log Keluar

Pengesahan Majikan

Kemaskini

Gaji Pokok Bulanan	RM 123
Gaji Bersih Bulanan	RM 456555

Hantar

ALAMAT

Peti Surat 127,
Bandar Kota Bharu
15710 Kota Bharu
Kelantan

HUBUNGI KAMI

Telefon: 09-1234567
Email: info@kada.gov.my

IKUTI KAMI

WAKTU OPERASI

Ahad - Rabu: 08:00 - 17:00
Khamis: 08:00 - 15:00
Jumaat: Rehat
Sabtu: Rehat

Hak Cipta © 2025 KADA. Semua Hak Terpelihara.

```

1 // Retrieve data from tb_loan for display loan details
2

```

```

3 $sql = "SELECT l_loanType, l_appliedLoan, l_loanPeriod,
4 l_monthlyInstalment, l_bankAccountNo, l_bankName,
5 l_monthlyGrossSalary, l_monthlyNetSalary, l_signature
6     FROM tb_loan
7     WHERE l_loanApplicationID = $loanApplicationID";
8 $result = mysqli_query($con, $sql);
9
10 // Retrieve data from tb_ltype for display loan name
11 if ($selected_jenis_pembiayaan_ID) {
12     $sql = "SELECT lt_desc
13         FROM tb_ltype
14         WHERE lt_lid = $selected_jenis_pembiayaan_ID";
15 }
16
17 // Retrieve data from tb_lbanks for display name bank
18 if ($selected_namaBank_ID) {
19     $sql = "SELECT lb_desc
20         FROM tb_lbanks
21         WHERE lb_id = $selected_namaBank_ID";
22     $result = mysqli_query($con, $sql);
23 }
24
25 // Retrieve data from tb_member for display member details
26 if ($memberNo !== null) {
27     $sql = "SELECT m_name, m_ic, m_email, m_gender, m_religion, m_race,
28 m_maritalStatus, m_pfNo, m_monthlySalary, m_homeAddress,
29 m_homePostcode, m_homeCity, m_homeState, m_positionGrade, m_position,
30 m_officeAddress, m_officePostcode, m_officeCity, m_officeState,
31 m_homeNumber, m_phoneNumber
32     FROM tb_member
33     WHERE m_memberNo = '$memberNo'";
34     $result = mysqli_query($con, $sql);
35 }
36
37 // Retrieve data from tb_ugender for display gender
38 if ($memberGender_ID) {
39     $sql = "SELECT ug_desc
40         FROM tb_ugender
41         WHERE ug_gid = $memberGender_ID";
42     $result = mysqli_query($con, $sql);
43 }
44
45 // Retrieve data from tb ureligion for display religion
46 if ($memberReligion_ID) {
47     $sql = "SELECT ua_desc
48         FROM tb ureligion
49         WHERE ua_rid = $memberReligion_ID";
50     $result = mysqli_query($con, $sql);
51 }
52
53 // Retrieve data from tb_urace for display race
54 if ($memberRace_ID) {
55     $sql = "SELECT ur_desc
56         FROM tb_urace
57         WHERE ur_rid = $memberRace_ID";
58     $result = mysqli_query($con, $sql);
59 }

```

```

60
61 // Retrieve data from tb_umaritalstatus for display marital status
62 if ($memberMaritalStatus_ID) {
63     $sql = "SELECT um_desc
64             FROM tb_umaritalstatus
65             WHERE um_mid = $memberMaritalStatus_ID";
66     $result = mysqli_query($con, $sql);
67 }
68
69 // Retrieve data from tb_homestate for display home state
70 if ($memberHomeState_ID) {
71     $sql = "SELECT st_desc
72             FROM tb_homestate
73             WHERE st_id = $memberHomeState_ID";
74     $result = mysqli_query($con, $sql);
75 }
76
77 // Retrieve data from tb_officestate for display office state
78 if ($memberOfficeState_ID) {
79     $sql = "SELECT st_desc
80             FROM tb_officestate
81             WHERE st_id = $memberOfficeState_ID";
82     $result = mysqli_query($con, $sql);
83 }
84
85 // Retrieve data from tb_guarantor for display guarantor 1 details
86 $sql = "SELECT g.g_memberNo, g.g_signature, m.m_name, m.m_ic,
87 m.m_pfNo
88         FROM tb_guarantor g
89         LEFT JOIN tb_member m ON g.g_memberNo = m.m_memberNo
90         WHERE g.g_guarantorID = '$guarantorID1'";
91 $result = mysqli_query($con, $sql);
92
93 if ($result && $row = mysqli_fetch_assoc($result)) {
94     $anggotaPenjamin1 = $row['g_memberNo'];
95     $signaturePenjamin1 = $row['g_signature'];
96     $namaPenjamin1 = $row['m_name'];
97     $sicPenjamin1 = $row['m_ic'];
98     $pfPenjamin1 = $row['m_pfNo'];
99 }
100
101 // Retrieve data from tb_guarantor for display guarantor 2 details
102 $sql2 = "SELECT g.g_memberNo, g.g_signature, m.m_name, m.m_ic,
103 m.m_pfNo
104         FROM tb_guarantor g
105         LEFT JOIN tb_member m ON g.g_memberNo = m.m_memberNo
106         WHERE g.g_guarantorID = '$guarantorID2'";
107 $result2 = mysqli_query($con, $sql2);
108
109 if ($result2 && $row2 = mysqli_fetch_assoc($result2)) {
110     $anggotaPenjamin2 = $row2['g_memberNo'];
111     $signaturePenjamin2 = $row2['g_signature'];
112     $namaPenjamin2 = $row2['m_name'];
113     $sicPenjamin2 = $row2['m_ic'];
114     $pfPenjamin2 = $row2['m_pfNo'];
115 }
```

Update loan details

Butir-Butir Pembiayaan

Jenis Pembiayaan
Al-Bai

Amaun Dipohon*
RM 1234

Tempoh Pembiayaan*
2 tahun

Ansuran Bulanan*
RM 56.56

*Sila rujuk jadual pembayaran balik pembiayaan skim

Nama Bank/Cawangan
Hong Leong Bank

Nama Bank/Cawangan
Hong Leong Bank

Bank Account
90876589

Gaji Kasar Bulanan
RM 123

Gaji Bersih Bulanan
RM 456555

Tandatangan
Choose File No file chosen

*Fail yang dibenarkan adalah dalam format PNG, JPG dan JPEG sahaja. Sila pastikan saiz fail tidak melebihi 5MB.

Simpan

Gaji Bersih Bulanan
RM 456555

Tandatangan
Choose File No file chosen

*Fail yang dibenarkan adalah dalam format PNG, JPG dan JPEG sahaja. Sila pastikan saiz fail tidak melebihi 5MB.

Simpan

ALAMAT
Peti Surat 127,
Bandar Kota Bharu
15710 Kota Bharu
Kelantan

HUBUNGI KAMI
Telefon: 09-1234567
Email: info@kada.gov.my

IKUTI KAMI

WAKTU OPERASI
Ahad - Rabu: 08:00 - 17:00
Khamis: 08:00 - 15:30
Jumaat: Rehat
Sabtu: Rehat

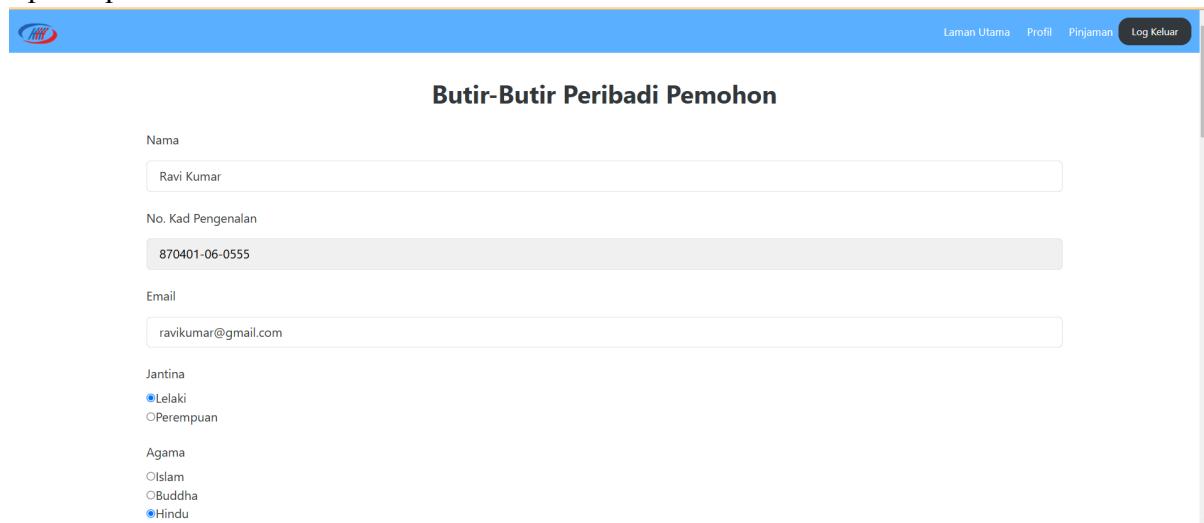
Hak Cipta © 2025 KADA. Semua Hak Terpelihara.

```

1 // Retrieve data from tb_policies to check for loan details does not
2 // exceed in tb_policies
3 $sql = "
4     SELECT * FROM tb_policies
5     ORDER BY p_policyID DESC
6     LIMIT 1;";
7 $result = mysqli_query($con, $sql);
8 $policy = mysqli_fetch_assoc($result);
9
10 // Modify data in tb_loan
11 if (!empty($loanApplicationID)) {
12     $sql = "UPDATE tb_loan
13         SET l_loanType = '$loanType',
14             l_appliedLoan = '$amaunDipohon',
15             l_loanPeriod = '$tempohPembayaran',
16             l_monthlyInstalment = '$ansuranBulanan',
17             l_loanPayable = '$tunggakan',
18             l_bankName = '$namaBank',
19             l_bankAccountNo = '$bankAcc',
20             l_monthlyGrossSalary = '$gajiKasar',
21             l_monthlyNetSalary = '$gajiBersih',
22             l_signature = '$fileSign'
23
24         WHERE l_loanApplicationID = '$loanApplicationID'";
25
26 }

```

Update personal details



The screenshot shows a web application interface for updating personal details. At the top, there is a blue header bar with a logo on the left and navigation links: Laman Utama, Profil, Pinjaman, and Log Keluar. The main content area has a title 'Butir-Butir Peribadi Pemohon'. Below the title, there are several input fields and dropdown menus:

- Nama:** A text input field containing 'Ravi Kumar'.
- No. Kad Pengenalan:** A text input field containing '870401-06-0555'.
- Email:** A text input field containing 'ravikumar@gmail.com'.
- Jantina:** A dropdown menu with options 'Lelaki' (selected) and 'Perempuan'.
- Agama:** A dropdown menu with options 'Islam' (selected), 'Buddha', and 'Hindu'.

Laman Utama Profil Pinjaman Log Keluar

Agama
 Islam
 Buddha
 Hindu
 Kristian
 Lain-lain: Nyatakan

Bangsa
 Melayu
 Cina
 India
 Lain-lain: Nyatakan

Taraf Perkahwinan
 Bujang
 Kahwin
 Cerai
 Kematian Pasangan

No. Anggota
3

Laman Utama Profil Pinjaman Log Keluar

No. Anggota
3

No. PF
1005

Alamat Rumah
No.25, Taman Desa Aman

Poskod
54000

Bandar
Kuala Lumpur

Negeri
WP Kuala Lumpur

Jawatan Gred

Laman Utama Profil Pinjaman Log Keluar

Jawatan Gred
J41

Jawatan
Pegawai Teknikal

Alamat Pejabat (Tempat Bertugas)
Jalan Dato Lundang

Poskod
15000

Bandar
Kota Bharu

Negeri
WP Kuala Lumpur

The screenshot shows a user profile page with the following fields:

- No. Telefon Rumah: 0182221234
- No. Telefon Bimbit: 0163344556
- Gaji Bulanan: RM 4500

A blue "Simpan" button is at the bottom left.

ALAMAT

Peti Surat 127,
Bandar Kota Bharu
15710 Kota Bharu
Kelantan

HUBUNGI KAMI

Telefon: 09-1234567
Email: info@kada.gov.my

IKUTI KAMI

[Facebook](#) [Twitter](#) [Instagram](#)

WAKTU OPERASI

Ahad - Rabu: 08:00 - 17:00
Khamis: 08:00 - 15:30
Jumaat: Rehat
Sabtu: Rehat

```

1 //Retrieve data from tb_member for checking details
2 if ($memberNo !== null) {
3     $sql = "SELECT m_name, m_ic, m_email, m_gender, m_religion, m_race,
4 m_maritalStatus, m_pfNo, m_monthlySalary, m_homeAddress,
5 m_homePostcode, m_homeCity, m_homeState, m_positionGrade, m_position,
6 m_officeAddress, m_officePostcode, m_officeCity, m_officeState,
7 m_homeNumber, m_phoneNumber
8         FROM tb_member
9         WHERE m_memberNo = '$memberNo'";
10    $result = mysqli_query($con, $sql);
11
12 // Modify data in tb_member
13 if (!empty($memberNo)) {
14     $sql = "UPDATE tb_member
15             SET m_name = '$nama',
16                 m_email = '$email',
17                 m_gender = '$selectedGender',
18                 m_religion = '$selectedReligion',
19                 m_race = '$selectedRace',
20                 m_maritalStatus = '$selectedMaritalRace',
21                 m_monthlySalary = '$selectedMonthlySalary',
22
23                 m_homeAddress = '$selectedHomeAdd',
24                 m_homePostcode = '$selectedHomePostcode',
25                 m_homeCity = '$selectedHomeCity',
26                 m_homeState = '$selectedHomeState',
27
28                 m_position = '$selectedPosition',
29                 m_positionGrade = '$selectedPositionGrade',
30
31                 m_officeAddress = '$selectedOfficeAdd',
32                 m_officePostcode = '$selectedOfficePostcode',
33                 m_officeCity = '$selectedOfficeCity',
34                 m_officeState = '$selectedOfficeState',
35
36                 m_phoneNumber = '$selectedNoTeleBim',
37                 m_homeNumber = '$selectedNoTelePej'
```

38	<code>WHERE m_memberNo = '\$memberNo'";</code>
39	
40	}

Update guarantor details

The screenshot displays two identical forms for updating guarantor details, one for 'Penjamin 1' and one for 'Penjamin 2'. Both forms are titled 'Butir-Butir Penjamin' and are part of a larger application interface.

Form Fields (Penjamin 1):

- No. Anggota: 2
- Nama: Mohd Hafiz
- No. Kad Pengenalan: 880701-05-0444
- No. PF: 1004
- Tandatangan: Choose File (No file chosen)

Form Fields (Penjamin 2):

- No. Anggota: 5
- Nama: Lim Wei Sheng
- No. Kad Pengenalan: 850301-08-0777
- No. PF: 1007
- Tandatangan: Choose File (No file chosen)

Common Application Header:

- Laman Utama
- Profil
- Pinjaman
- Log Keluar

Note: *Fail yang dibenarkan adalah dalam format PNG, JPG, dan JPEG sahaja. Sila pastikan saiz fail tidak melebihi 5MB.

No. PF
1007

Tandatangan
Choose File No file chosen

*Fail yang dibenarkan adalah dalam format PNG, JPG, dan JPEG sahaja. Sila pastikan saiz fail tidak melebihi 5MB.

ALAMAT
Peti Surat 127,
Bandar Kota Bharu
15710 Kota Bharu
Kelantan

HUBUNGI KAMI
Telefon: 09-1234567
Email: info@kada.gov.my

WAKTU OPERASI
Ahad - Rabu: 08:00 - 17:00
Khamis: 08:00 - 15:30
Jumaat: Rehat
Sabtu: Rehat

IKUTI KAMI

Hak Cipta © 2025 KADA. Semua Hak Terpelihara.

```

1 //Retrieve data from tb_guarantor for guarantor 1 details
2 if ($loanApplicationID !== null) {
3     $sql = "SELECT g_memberNo, g_signature, m.m_name, m.m_ic, m.m_pfNo
4         FROM tb_guarantor g
5             LEFT JOIN tb_member m ON g.g_memberNo = m.m_memberNo
6                 WHERE g.g_loanApplicationID = '$loanApplicationID'";
7     $result = mysqli_query($con, $sql);
8 }
9
10 //Retrieve data from tb_guarantor for guarantor 2 details
11 if ($guarantorID2 !== null) {
12     $sql = "SELECT g_memberNo, g_signature, m.m_name, m.m_ic, m.m_pfNo
13         FROM tb_guarantor g
14             LEFT JOIN tb_member m ON g.g_memberNo = m.m_memberNo
15                 WHERE g.g_loanApplicationID = '$loanApplicationID' AND
16 g.g_guarantorID = '$guarantorID2'";
17     $result = mysqli_query($con, $sql);
18 }
19
20 // Modify data in tb_guarantor for guarantor 1
21 if (!empty($memberNo1)) {
22     $update_stmt1 = mysqli_prepare
23     ($con, "UPDATE tb_guarantor
24         SET g_memberNo = ?, g_signature = ?
25         WHERE g_guarantorID = ?");
26     mysqli_stmt_bind_param($update_stmt1, "sss", $memberNo1,
27 $fileSignPenjamin1, $guarantorID1);
28     if (!mysqli_stmt_execute($update_stmt1)) {
29         throw new Exception("Failed to update guarantor 1: " .
30 mysqli_stmt_error($update_stmt1));
31     }
32     mysqli_stmt_close($update_stmt1);
33 }
34
35 // Modify data in tb_guarantor for guarantor 2
36 if (!empty($memberNo2)) {
37     $update_stmt2 = mysqli_prepare

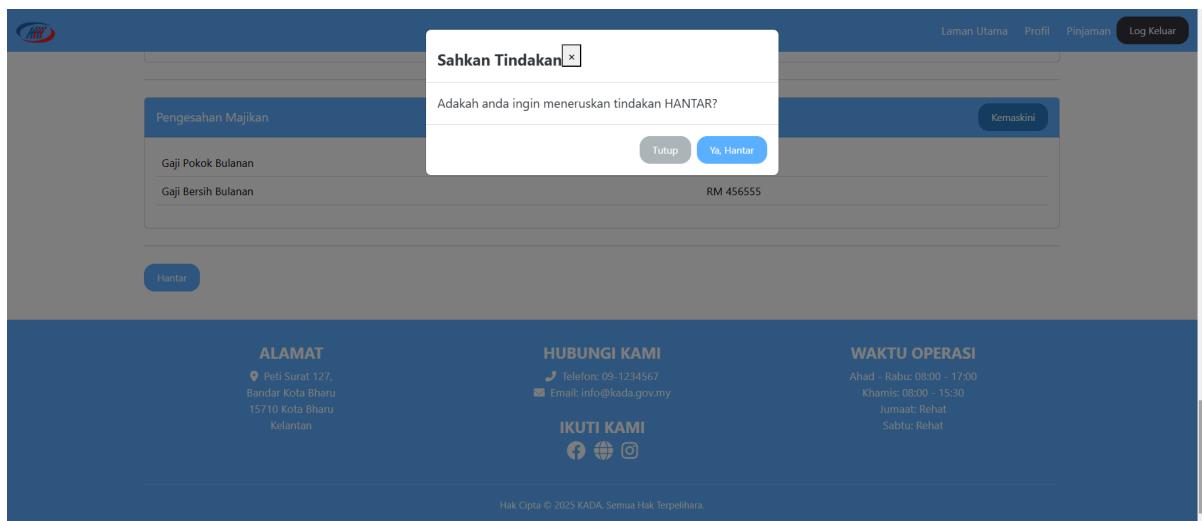
```

```

38      ($con, "UPDATE tb_guarantor
39          SET g_memberNo = ?, g_signature = ?
40          WHERE g_guarantorID = ?");
41      mysqli_stmt_bind_param($update_stmt2, "sss", $memberNo2,
42      $fileSignPenjamin2, $guarantorID2);
43      if (!mysqli_stmt_execute($update_stmt2)) {
44          throw new Exception("Failed to update guarantor 2: " .
45      mysqli_stmt_error($update_stmt2));
46      }
47      mysqli_stmt_close($update_stmt2);
48  }
49
50

```

Confirmation for submit



Use Case: View Financial Status

Member Main Page

Maklumat Saham Ahli		Rekod Transaksi
Modah Syer	RM 5.00	
Modal Yuran	RM 50.00	
Simpanan Tetap	RM 350.00	

The screenshot shows the KADA mobile application interface. At the top, there is a navigation bar with icons for home, profile, loans, and logout. Below the navigation bar, there are two tables showing financial statements:

	RM 50.00
Tabung Anggota	
Simpanan Anggota	RM 400.00

Maklumat Pinjaman Ahli	
Rekod Transaksi	
Al-Bai	RM 110.00
Al-Innah	RM 0.00
Al-Qadrul Hassan	RM 0.00
Baik Pulih Kenderaan	RM 0.00
Karnival Musim Istimewa	RM 0.00
Khas	RM 0.00
Road Tax dan Insurans	RM 0.00

At the bottom of the screen, there is a blue footer section containing contact information:

- ALAMAT**: Peti Surat 127, Bandar Kota Bharu 15/10 Kota Bharu Kelantan
- HUBUNGI KAMI**: Telefon: 09-1234567 Email: info@kada.gov.my
- IKUTI KAMI**: Facebook, Instagram, YouTube
- WAKTU OPERASI**: Ahad - Rabu: 08:00 - 17:00
Khamis: 08:00 - 15:30
Jumaat: Rehat
Sabtu: Rehat

Hak Cipta © 2025 KADA. Semua Hak Terpelihara.

```

1 //Retrieve data from tb_financial for member share information
2 SELECT * FROM tb_financial
3           WHERE tb_financial.f_memberNo = '$u_id';
4
5 //Retrieve data from tb_loan and tb_ltype for member loan information
6 SELECT
7           tb_ltype.lt_desc AS loanName,
8           COALESCE(SUM(tb_loan.l_loanPayable)) AS totalLoan
9           FROM tb_ltype
10          LEFT JOIN tb_loan
11            ON tb_loan.l_loanType=tb_ltype.lt_lid
12            AND tb_loan.l_memberNo=' $u_id'
13            AND tb_loan.l_status=3
14            GROUP BY tb_ltype.lt_desc
15            ORDER BY tb_ltype.lt_desc;
16
17 //Retrieve data from tb_banner for advertisement
18 SELECT * FROM tb_banner;
19

```

Salary Deduction Record Page



Laman Utama Profil Pinjaman Log Keluar

Rekod Potongan Gaji

1 2 >

Rekod Potongan Gaji

Kembali

ID	Jenis	Amaun (RM)	Bulan	Tahun	Tarikh Direkod
3	Tabung Anggota	350.00	Januari	2025	26-11-2025
23	Modah Syer	50.00	Januari	2025	10-01-2025
24	Modal Yuran	50.00	Januari	2025	10-01-2025
25	Simpanan Tetap	50.00	Januari	2025	10-01-2025
15	Simpanan Anggota	50.00	Januari	2025	07-01-2025
13	Simpanan Tetap	50.00	Januari	2025	06-01-2025
14	Tabung Anggota	50.00	Januari	2025	06-01-2025
26	Tabung Anggota	50.00	Januari	2025	06-01-2025
12	Modal Yuran	50.00	Januari	2025	05-01-2025
27	Simpanan Anggota	50.00	Januari	2025	05-01-2025

ALAMAT

❖ Peti Surat 127,
Bandar Kota Bharu
15710 Kota Bharu
Kelantan

HUBUNGI KAMI

📞 Telefon: 09-1234567
✉️ Email: Info@kada.gov.my

IKUTI KAMI



WAKTU OPERASI

Ahad - Rabu: 08:00 - 17:00
Khamis: 08:00 - 15:30
Jumaat: Rehat
Sabtu: Rehat

Hak Cipta © 2025 KADA. Semua Hak Terpelihara.

```

1 //Count number of desired records from tb_transaction for member
2 salary deduction record page pagination
3 SELECT COUNT(*) AS total FROM tb_transaction
4           WHERE t_memberNo='$_id' AND t_transactionType IN
5 (1,2,3,4,5);
6
7 //Retrieve data from tb_transaction, tb_ttype, and tb_rmonth for
8 member salary deduction record
9 SELECT tb_transaction.*,
10           tb_ttype.tt_desc AS type,
11           tb_rmonth.rm_desc AS month
12      FROM tb_transaction
13      LEFT JOIN tb_ttype ON
14 tb_transaction.t_transactionType=tb_ttype.tt_id
15      LEFT JOIN tb_rmonth ON tb_transaction.t_month=tb_rmonth.rm_id
16      WHERE tb_transaction.t_memberNo = '$_id' AND
17 tb_transaction.t_transactionType IN (1,2,3,4,5)
18      ORDER BY tb_transaction.t_transactionDate DESC
19      LIMIT $records_per_page OFFSET $offset;
20

```

Loan Repayment Record Page

Rekod Bayaran Balik

ID	Jenis	Amaun (RM)	Bulan	Tahun	Tarikh Direkod
9	Khas	150.00	Disember	2024	01-12-2025
20	Khas	50.00	Januari	2025	09-01-2025
21	Karnival Musim Istimewa	50.00	Januari	2025	09-01-2025
22	Al-Qadruh Hassan	50.00	Januari	2025	09-01-2025
18	Baik Pulih Kenderaan	50.00	Januari	2025	08-01-2025
19	Road Tax and Insurance	50.00	Januari	2025	08-01-2025
16	Al-Bai	50.00	Januari	2025	07-01-2025
17	Al-Innah	50.00	Januari	2025	07-01-2025

ALAMAT
Peti Surat 127,
Bandar Kota Bharu
15710 Kota Bharu
Kelantan

HUBUNGI KAMI
Telefon: 09-1234567
Email: info@kada.gov.my

WAKTU OPERASI
Ahad - Rabu: 08:00 - 17:00
Khamis: 08:00 - 15:30
Jumaat: Rehat
Sabtu: Rehat

Hak Cipta © 2025 KADA. Semua Hak Terpelihara.

```

1 //Count number of desired records from tb_transaction for member loan
2 repayment record page pagination
3 SELECT COUNT(*) AS total FROM tb_transaction
4     WHERE t_memberNo='$_id' AND t_transactionType IN
5 (6,7,8,9,10,11,12);
6
7 //Retrieve data from tb_transaction, tb_ttype, and tb_rmonth for
8 member loan repayment transaction records
9 SELECT tb_transaction.*,
10         tb_ttype.tt_desc AS type,
11         tb_rmonth.rm_desc AS month
12     FROM tb_transaction
13     LEFT JOIN tb_ttype ON
14 tb_transaction.t_transactionType=tb_ttype.tt_id
15     LEFT JOIN tb_rmonth ON tb_transaction.t_month=tb_rmonth.rm_id
16     WHERE tb_transaction.t_memberNo = '$_id' AND
17 tb_transaction.t_transactionType IN (6,7,8,9,10,11,12)
18     ORDER BY tb_transaction.t_transactionDate DESC
19     LIMIT $records_per_page OFFSET $offset;
20

```

Use Case: Update User Profile

Member Profile

Laman Utama Profil Pinjaman Log Keluar

Maklumat Peribadi

Butir-butir Peribadi Pemohon	
Nama	Ravi Kumar
No. KP	870401-06-0555
E-mel	ravikumar@gmail.com
Jantina	Lelaki
Agama	Hindu
Bangsa	India
Taraf Perkahwinan	Kahwin
Alamat Rumah	No.25, Taman Desa Aman
Poskod	54000
Bandar	Kuala Lumpur
Negeri	WP Kuala Lumpur
No. Anggota	3

Laman Utama Profil Pinjaman Log Keluar

No. PF	1005
Jawatan	Pegawai Teknikal
Gred	J41
Alamat Pejabat	Jalan Dato Lundang
Poskod	15000
Bandar	Kota Bharu
Negeri	WP Kuala Lumpur
No. Tel Bimbit	0163344556
No. Tel Rumah	0182221234
Gaji Bulanan	4500
Nama Bank	Al Rajhi Bank Malaysia
No. Akaun Bank	1234567893

Maklumat Keluarga dan Pewaris

Keluarga / Pewaris 1

Kemaskini

Laman Utama Profil Pinjaman Log Keluar

Nama	Rohan Kumar
No. KP	920301-04-0333
Hubungan	Anak
Keluarga / Pewaris 2	
Nama	Lakshmi Kumar
No. KP	850301-08-0777
Hubungan	Saudara kandung
Keluarga / Pewaris 3	
Nama	Ravi Kumar
No. KP	800201-08-0444
Hubungan	Anak
Keluarga / Pewaris 4	
Nama	aaa
No. KP	123456789

Laman Utama Profil Pinjaman Log Keluar

Hubungan	Lain-lain
Keluarga / Pewaris 5	
Nama	CC
No. KP	999888777
Hubungan	Anak
Keluarga / Pewaris 6	
Nama	s
No. KP	126597738
Hubungan	Anak

ALAMAT
Peti Surat 127,
Bandar Kota Bharu
15710 Kota Bharu
Kelantan

HUBUNGI KAMI
Telefon: 09-1234567
Email: info@kada.gov.my

IKUTI KAMI
[Facebook](#) [Twitter](#) [Instagram](#)

WAKTU OPERASI
Ahad - Rabu: 08:00 - 17:00
Khamis: 08:00 - 15:30
Jumaat: Rehat
Sabtu: Rehat

Hak Cipta © 2025 KADA. Semua Hak Terpelihara.

```

1 //Retrieve data from tb_member, tb_ugender, tb_urace, tb_ureligion,
2 tb_umaritalstatus, tb_homeState, and tb_officeState for member
3 personal information
4 SELECT tb_member.*,
5         tb_ugender.ug_desc AS gender,
6         tb_urace.ur_desc AS race,
7         tb_ureligion.ua_desc AS religion,
8         tb_umaritalstatus.um_desc AS maritalstatus,
9         tb_homeState.st_desc AS homeState,
```

```

10          tb_officeState.st_desc AS officeState,
11          tb_member.m_memberApplicationID
12      FROM tb_member
13      LEFT JOIN tb_ugender ON tb_member.m_gender=tb_ugender.ug_gid
14      LEFT JOIN tb_urace ON tb_member.m_race=tb_urace.ur_rid
15      LEFT JOIN tb_ureligion ON
16      tb_member.m_religion=tb_ureligion.ua_rid
17      LEFT JOIN tb_umaritalstatus ON
18      tb_member.m_maritalStatus=tb_umaritalstatus.um_mid
19      LEFT JOIN tb_homeState ON
20      tb_member.m_homeState=tb_homeState.st_id
21      LEFT JOIN tb_officeState ON
22      tb_member.m_officeState=tb_officeState.st_id
23      WHERE tb_member.m_memberNo = '$u_id';
24
25 //Retrieve data from tb_heir and tb_hrelation for member's heirs
26 information
27 SELECT tb_heir.* ,
28         tb_hrelation.hr_desc AS heirrelation
29         FROM tb_heir
30         LEFT JOIN tb_hrelation ON tb_heir.h_relationWithMember =
31      tb_hrelation.hr_rid
32         WHERE h_memberApplicationID = '$memberApplicationID'
33 ORDER BY h_heirID;
34

```

Update Member Personal Information



The screenshot shows a web-based application interface for updating member personal information. At the top, there is a blue header bar with a logo on the left and navigation links: Laman Utama, Profil, Pinjaman, and Log Keluar. Below the header, the main content area has a title "Butir-butir Peribadi Pemohon". The form fields include:

- Nama ***: A text input field containing "Ravi Kumar".
- No. Kad Pengenalan**: A text input field containing "870401-06-0555".
- Tarikh Lahir**: A date input field showing "01-04-1987".
- Jantina**: Radio button options for gender. "Lelaki" is selected.
- Agama ***: Radio button options for religion. "Hindu" is selected.
- Bangsa ***: Radio button options for ethnicity. "India" is selected.

Laman Utama Profil Pinjaman Log Keluar

Taraf Perkahwinan *

Bujang Kahwin Cerai Kematiān Pasangan

E-mel *

ravikumar@gmail.com

Alamat Rumah *

No.25, Taman Desa Aman

Poskod * Bandar *

54000 Kuala Lumpur

Negeri *

WP Kuala Lumpur

No. Anggota No. PF

3 1005

Laman Utama Profil Pinjaman Log Keluar

Jawatan *

Pegawai Teknikal

Gred *

J41

Alamat Pejabat (Tempat Bertugas) *

Jalan Dato Lundang

Poskod * Bandar *

15000 Kota Bharu

Negeri *

WP Kuala Lumpur

No. Telefon Rumah No. Telefon Bimbit *

0182221234 0163344556

Laman Utama Profil Pinjaman Log Keluar

Gaji Bulanan (RM) *

4500

Nama Bank

Al Rajhi Bank Malaysia

No. Akaun Bank

1234567893

Simpan

ALAMAT
Peti Surat 127,
Bandar Kota Bharu
15710 Kota Bharu
Kelantan

HUBUNGI KAMI
Telefon: 09-1234567
Email: info@kada.gov.my

WAKTU OPERASI
Ahad - Rabu: 08:00 - 17:00
Khamis: 08:00 - 15:30
Jumaat: Rehat
Sabtu: Rehat

Hak Cipta © 2025 KADA. Semua Hak Terpelihara.

```

1 //Retrieve data from tb_member, tb_ugender, tb_urace, tb_ureligion,
2 tb_umaritalstatus, tb_homeState, and tb_officeState for updating
3 member personal information
4 SELECT tb_member.*,
5         tb_ugender.ug_desc AS gender,
6         tb_urace.ur_desc AS race,
7         tb_ureligion.ua_desc AS religion,
8         tb_umaritalstatus.um_desc AS maritalstatus,
9         tb_homeState.st_desc AS homeState,
10        tb_officeState.st_desc AS officeState,
11        tb_member.m_memberApplicationID
12    FROM tb_member
13    LEFT JOIN tb_ugender ON tb_member.m_gender=tb_ugender.ug_gid
14    LEFT JOIN tb_urace ON tb_member.m_race=tb_urace.ur_rid
15    LEFT JOIN tb_ureligion ON
16    tb_member.m_religion=tb_ureligion.ua_rid
17    LEFT JOIN tb_umaritalstatus ON
18    tb_member.m_maritalStatus=tb_umaritalstatus.um_mid
19    LEFT JOIN tb_homeState ON
20    tb_member.m_homeState=tb_homeState.st_id
21    LEFT JOIN tb_officeState ON
22    tb_member.m_officeState=tb_officeState.st_id
23        WHERE tb_member.m_memberNo = '$u_id';
24
25 //Update member personal information
26 UPDATE tb_member
27     SET
28     m_name = '$name',
29     m_religion = '$agama',
30     m_race = '$race',
31     m_maritalStatus = '$maritalstatus',
32     m_email = '$emel',
33     m_homeAddress = '$homeaddress',
34     m_homePostcode = '$homepostcode',
35     m_homeCity = '$homecity',
36     m_homeState = '$homestate',
37     m_position = '$position',
38     m_positionGrade = '$grade',
39     m_officeAddress = '$officeaddress',
40     m_officePostcode = '$officepostcode',
41     m_officeCity = '$officecity',
42     m_officeState = '$officestate',
43     m_faxNumber = '$faxphonenum',
44     m_phoneNumber = '$phonenum',
45     m_homeNumber = '$homephonenum',
46     m_monthlySalary = '$monthlysalary',
47     WHERE m_memberNo='$u_id';
48

```

Update Member's Heirs Information



Laman Utama Profil Pinjaman Log Keluar

Maklumat Keluarga dan Pewaris

Maklumat Pewaris 1

Nama *

Rohan Kumar

No. KP / No. Srt Beranak *

920301-04-0333

Hubungan *

Anak

Padam Pewaris

Maklumat Pewaris 2

Nama *

Lakshmi Kumar

Laman Utama Profil Pinjaman Log Keluar

No. KP / No. Srt Beranak *

850301-08-0777

Hubungan *

Saudara Kandung

Padam Pewaris

Maklumat Pewaris 3

Nama *

Ravi Kumar

No. KP / No. Srt Beranak *

800201-08-0444

Hubungan *

Anak

Padam Pewaris

Laman Utama Profil Pinjaman Log Keluar

Maklumat Pewaris 4

Nama *

No. KP / No. Srt Beranak *

Hubungan *

Padam Pewaris

Maklumat Pewaris 5

Nama *

No. KP / No. Srt Beranak *

Hubungan *

Padam Pewaris

Maklumat Pewaris 6

Nama *

No. KP / No. Srt Beranak *

Hubungan *

Padam Pewaris
Tambah Pewaris
Simpan
Kembali

Laman Utama Profil Pinjaman Log Keluar

ALAMAT
📍 Peti Surat 127,
Bandar Kota Bharu
15710 Kota Bharu
Kelantan

HUBUNGI KAMI
📞 Telefon: 09-1234567
✉️ Email: info@kada.gov.my

IKUTI KAMI
Facebook
Instagram
Twitter

Hak Cipta © 2025 KADA. Semua Hak Terpelihara.

WAKTU OPERASI
Ahad - Rabu: 08:00 - 17:00
Khamis: 08:00 - 15:30
Jumaat: Rehat
Sabtu: Rehat

```

1 //Retrieve data from tb_member, tb_ugender, tb_urace, tb_ureligion,
2 tb_umaritalstatus, tb_homeState, and tb_officeState for member
3 personal information
4 SELECT tb_member.*,
5           tb_ugender.ug_desc AS gender,
6           tb_urace.ur_desc AS race,

```

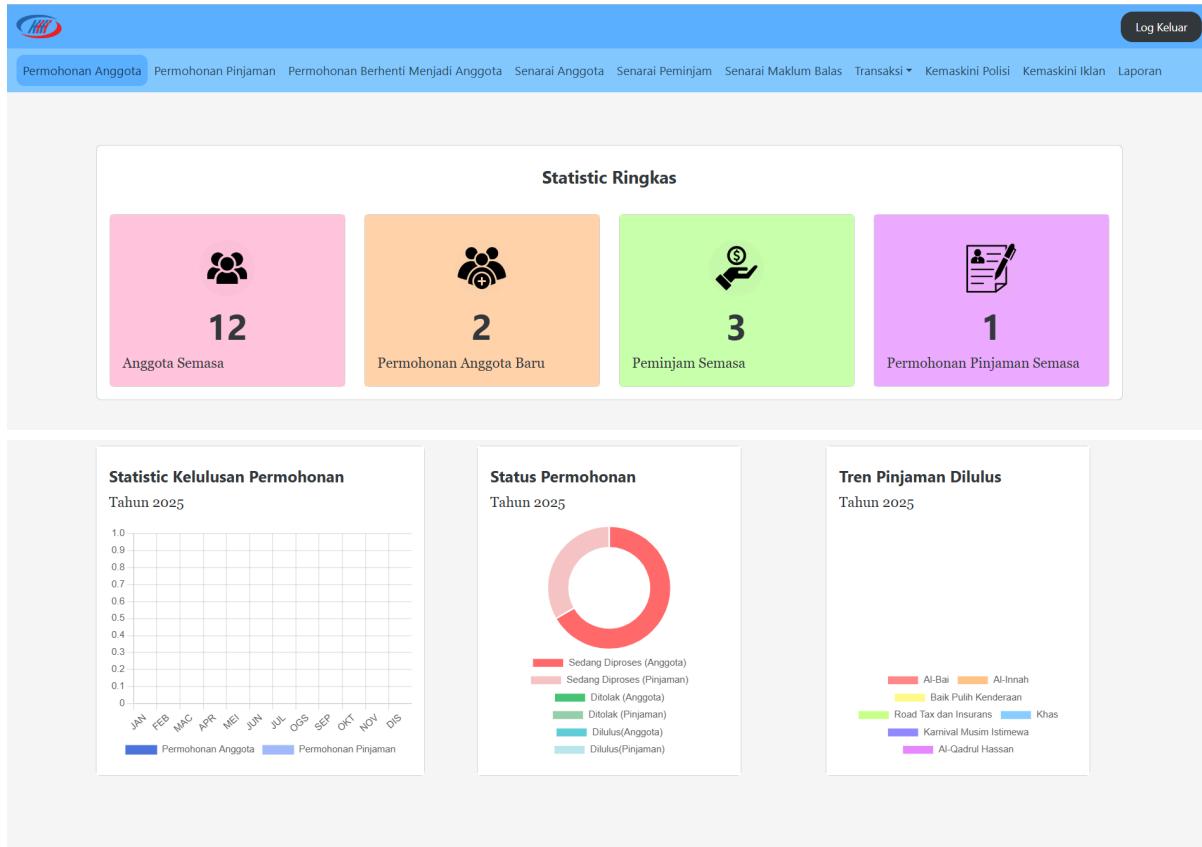
```

7      tb_ureligion.ua_desc AS religion,
8      tb_umaritalstatus.um_desc AS maritalstatus,
9      tb_homeState.st_desc AS homeState,
10     tb_officeState.st_desc AS officeState,
11     tb_member.m_memberApplicationID
12   FROM tb_member
13   LEFT JOIN tb_ugender ON
14 tb_member.m_gender=tb_ugender.ug_gid
15   LEFT JOIN tb_urace ON tb_member.m_race=tb_urace.ur_rid
16   LEFT JOIN tb_ureligion ON
17 tb_member.m_religion=tb_ureligion.ua_rid
18   LEFT JOIN tb_umaritalstatus ON
19 tb_member.m_maritalStatus=tb_umaritalstatus.um_mid
20   LEFT JOIN tb_homeState ON
21 tb_member.m_homeState=tb_homeState.st_id
22   LEFT JOIN tb_officeState ON
23 tb_member.m_officeState=tb_officeState.st_id
24   WHERE tb_member.m_memberNo = '$u_id';
25
26 //Retrieve data from tb_heir, and tb_hrelation for updating member's
27 heirs information
28 SELECT tb_heir.*, tb_hrelation.hr_desc AS heirrelation
29   FROM tb_heir
30   LEFT JOIN tb_hrelation ON tb_heir.h_relationWithMember =
31 tb_hrelation.hr_rid
32   WHERE h_memberApplicationID = '$memberApplicationID'
33   ORDER BY h_heirID;
34
35 //Update heir information
36 UPDATE tb_heir SET h_name = '$heirName', h_ic = '$heirIC',
37 h_relationWithMember = '$heirRelation'
38   WHERE h_heirID = '$heirID';
39
40 SELECT * FROM tb_heir WHERE h_memberApplicationID =
41 '$memberApplicationID' AND h_ic = '$heirIC'
42
43 //Add new heir
44 INSERT INTO tb_heir (h_memberApplicationID, h_name, h_ic,
45 h_relationWithMember)
46   VALUES ('$memberApplicationID', '$heirName',
47 '$heirIC', '$heirRelation');
48
49 //Delete existing heir
50 DELETE FROM tb_heir WHERE h_heirID = '$heirId';
51
52
53
54
55
56

```

4.1.2 Admin View

Admin Dashboard



```

1 //Count number of desired records from tb_member and tb_loan for
2 admin dashboard
3 SELECT COUNT(*) AS total_members FROM tb_member WHERE m_status='3';
4
5 SELECT COUNT(*) AS pending_members FROM tb_member WHERE m_status='1';
6
7 SELECT COUNT(DISTINCT l_memberNo) AS loan_borrowers FROM tb_loan
8 WHERE l_status='3';
9
10 SELECT COUNT(*) AS pending_loans FROM tb_loan WHERE l_status='1';
11
12 SELECT COUNT(*) AS count FROM tb_member WHERE m_approvalDate BETWEEN
13 '$start_date' AND '$end_date' AND m_status='3';
14
15 SELECT COUNT(*) AS count FROM tb_loan WHERE l_approvalDate BETWEEN
16 '$start_date' AND '$end_date' AND l_status='3';
17
18 SELECT COUNT(*) AS pending_member_applications FROM tb_member WHERE
19 m_status='1' AND m_applicationDate BETWEEN '$currentYear-01-01' AND
20 '$currentYear-12-31';
21
22

```

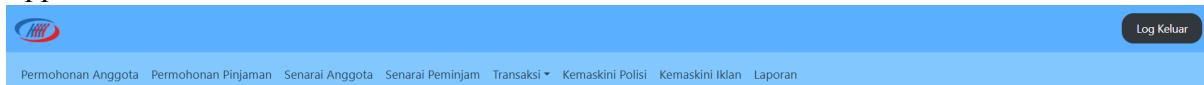
```

23 SELECT COUNT(*) AS rejected_member_applications FROM tb_member WHERE
24 m_status='2' AND m_approvalDate BETWEEN '$currentYear-01-01' AND
25 '$currentYear-12-31';
26
27 SELECT COUNT(*) AS approved_member_applications FROM tb_member WHERE
28 m_status='3' AND m_approvalDate BETWEEN '$currentYear-01-01' AND
29 '$currentYear-12-31';
30
31 SELECT COUNT(*) AS pending_loan_applications FROM tb_loan WHERE
32 l_status='1' AND l_applicationDate BETWEEN '$currentYear-01-01' AND
33 '$currentYear-12-31';
34
35 SELECT COUNT(*) AS rejected_loan_applications FROM tb_loan WHERE
36 l_status='2' AND l_approvalDate BETWEEN '$currentYear-01-01' AND
37 '$currentYear-12-31';
38
39 SELECT COUNT(*) AS approved_loan_applications FROM tb_loan WHERE
40 l_status='3' AND l_approvalDate BETWEEN '$currentYear-01-01' AND
41 '$currentYear-12-31';
42
43 SELECT COUNT(*) AS al_bai_loan_applications FROM tb_loan WHERE
44 l_loanType='1' AND l_status='3' AND l_approvalDate BETWEEN
45 '$currentYear-01-01' AND '$currentYear-12-31';
46
47 SELECT COUNT(*) AS al_innah_loan_applications FROM tb_loan WHERE
48 l_loanType='2' AND l_status='3' AND l_approvalDate BETWEEN
49 '$currentYear-01-01' AND '$currentYear-12-31';
50
51 SELECT COUNT(*) AS kenderaan_loan_applications FROM tb_loan WHERE
52 l_loanType='3' AND l_status='3' AND l_approvalDate BETWEEN
53 '$currentYear-01-01' AND '$currentYear-12-31';
54
55 SELECT COUNT(*) AS road_tax_loan_applications FROM tb_loan WHERE
56 l_loanType='4' AND l_status='3' AND l_approvalDate BETWEEN
57 '$currentYear-01-01' AND '$currentYear-12-31';
58
59 SELECT COUNT(*) AS khas_loan_applications FROM tb_loan WHERE
60 l_loanType='5' AND l_status='3' AND l_approvalDate BETWEEN
61 '$currentYear-01-01' AND '$currentYear-12-31';
62
63 SELECT COUNT(*) AS karnival_loan_applications FROM tb_loan WHERE
64 l_loanType='6' AND l_status='3' AND l_approvalDate BETWEEN
65 '$currentYear-01-01' AND '$currentYear-12-31';
66
67 SELECT COUNT(*) AS al_qadrul_loan_applications FROM tb_loan WHERE
68 l_loanType='7' AND l_status='3' AND l_approvalDate BETWEEN
69 '$currentYear-01-01' AND '$currentYear-12-31';
70

```

Use Case: Approve Membership

Approval Dashboard



The screenshot shows a web-based approval dashboard. At the top, there is a navigation bar with links: Permohonan Anggota, Permohonan Pinjaman, Senarai Anggota, Senarai Peminjaman, Transaksi, Kemaskini Polisi, Kemaskini Iklan, and Laporan. On the far right of the navigation bar is a "Log Keluar" button. Below the navigation bar, the main content area has a title "Permohonan Anggota". A table follows, displaying five rows of member application data. The columns are labeled: No. Aplikasi, No. PF, Nama Anggota, Tarikh Pohon, and Butiran. Each row contains a "..." link under the "Butiran" column. The first row shows application ID 1009, member ID 1009, name Farid Hakim, date 09-11-2024, and a "..." link. The second row shows application ID 1010, member ID 1010, name Zarina Mohd, date 10-11-2024, and a "..." link. The third row shows application ID 1011, member ID 1011, name Shalini Devi, date 11-11-2024, and a "..." link. The fourth row shows application ID 1012, member ID 1012, name Roslan Mansor, date 12-11-2024, and a "..." link.

Permohonan Anggota

No. Aplikasi	No. PF	Nama Anggota	Tarikh Pohon	Butiran
1009	1009	Farid Hakim	09-11-2024	...
1010	1010	Zarina Mohd	10-11-2024	...
1011	1011	Shalini Devi	11-11-2024	...
1012	1012	Roslan Mansor	12-11-2024	...

1

```
1 // Select the member with the status ("Sedang Diproses")
2 "SELECT
3     tb_member.m_memberApplicationID,
4     tb_member.m_pfNo,
5     tb_member.m_name,
6     DATE_FORMAT(tb_member.m_applicationDate, '%d-%m-%Y') AS
7     formattedDate
8     FROM tb_member
9     WHERE m_status = 1";
10
11
12
13
14
15 // Get total records count for pagination
16 "SELECT COUNT(*) FROM tb_member WHERE m_status = 1";
```

Member Detail

Maklumat Peribadi Pemohon	
No. Aplikasi Anggota	1014
No. PF	1016
Nama Anggota	Ali
No. Kad Pengenalan	040404-04-0404
Jantina	Lelaki
Agama	Buddha
Bangsa	Cina
Status Perkahwinan	Cerai
Alamat Rumah	10, Jalan Ali
No. Telefon	0123456789
Email	ali@gmail.com
No. Telefon Rumah	N/A
Jawatan	Pegawai
Gred	M14
Alamat Pejabat	11
Gaji Bulanan	10000
Tarikh Pohon:	13-01-2025 01:14:45

Maklumat Saham Pemohon	
Fee Masuk	35
Modal Yuran	35
Deposit	50
alAbras	35
Simpanan Tetap	50
Fee Lain	50

Status Anggota	
<input style="width: 100%;" type="button" value="Pilih Status"/>	
<input style="width: 100px; margin-right: 10px;" type="button" value="Kembali"/> <input style="width: 100px;" type="button" value="Hantar"/>	

```

1 // Select the member and display the details
2 "SELECT * FROM tb_member
3     LEFT JOIN tb_status ON tb_member.m_status = tb_status.s_sid
4     LEFT JOIN tb_ugender ON tb_member.m_gender = tb_ugender.ug_gid
5     LEFT JOIN tb_ureligion ON tb_member.m_religion =
6         tb_ureligion.ua_rid
7     LEFT JOIN tb_urace ON tb_member.m_race = tb_urace.ur_rid
8     LEFT JOIN tb_umaritalstatus ON tb_member.m_maritalStatus =
9         tb_umaritalstatus.um_mid
10    WHERE tb_member.m_memberApplicationID = '$mApplicationID'";
11
12
13

```

View Member List After Approval



No. Anggota	No. PF	Nama Anggota	Tarikh Masuk	Butiran
1	1	Tan Mei Ling	14-11-2025	...
2	1004	Mohd Hafiz	14-11-2025	...
3	1005	Ravi Kumar	14-11-2025	...
4	1006	Anisa Karim	14-11-2025	...
5	1007	Lim Wei Sheng	14-11-2025	...
6	1008	Aminah Saad	14-11-2024	...
8	1009	Farid Hakim		...
9	1010	Zarina Mohd		...
7	1013	Ali yang bernama Ali	20-01-2025	...

1

```

1 // Select SQL query with DATE_FORMAT for m_approvalDate to
2 // display in a list
3
4 $sql = "SELECT m_memberNo, m_pfNo, m_name,
5         DATE_FORMAT(m_approvalDate, '%d-%m-%Y') AS formattedDate
6         FROM tb_member
7         WHERE m_status = 3";
8

```

View the details of the member

Maklumat Peribadi Ahli	
No. Aplikasi Anggota	1013
No. Anggota	7
No. PF	1013
Nama Anggota	Ali yang bernama Ali
No. Kad Pengenalan	045689-01-5698
Jantina	Lelaki
Agama	Islam
Bangsa	Cina
Status Perkahwinan	Cerai
Alamat Rumah	10, Jalan
No. Telefon	1452236541
Email	bello@graduate.utm.my
No. Telefon Rumah	N/A
Jawatan	Manager
Gred	N17
Alamat Pejabat	10, Ka
Gaji Bulanan	8500
Tarikh Pohon:	2025-01-20 08:32:49
Tarikh Lulus:	2025-01-20 08:34:25

Maklumat Saham Ahli	
Fee Masuk	40
Modal Yuran	50
Deposit	50
alAbrar	5
Simpanan Tetap	0
Fee Lain	0

Kembali

```

1 // Display the member details
2 "SELECT * FROM tb_member
3     LEFT JOIN tb_ugender ON tb_member.m_gender =
4 tb_ugender.ug_gid
5     LEFT JOIN tb_ureligion ON tb_member.m_religion =
6 tb_ureligion.ua_rid
7     LEFT JOIN tb_urace ON tb_member.m_race = tb_urace.ur_rid
8     LEFT JOIN tb_umaritalstatus ON tb_member.m_maritalStatus
9 = tb_umaritalstatus.um_mid
10
11
12
13
14
15 WHERE m_memberNo = '$member_id'";
16

```

Use Case: Approve Loan Application

Approval Dashboard

The screenshot shows a web-based application interface. At the top, there is a blue header bar with the logo 'MFI' on the left and a 'Log Keluar' button on the right. Below the header, a navigation menu includes links for 'Permohonan Anggota', 'Permohonan Pinjaman', 'Senarai Anggota', 'Senarai Peminjam', 'Transaksi', 'Kemaskini Polisi', 'Kemaskini Iklan', and 'Laporan'. The main content area is titled 'Permohonan Pinjaman' and displays a table with two rows of data. The columns are labeled 'No. Pinjaman', 'No. Anggota', 'Nama Anggota', 'Tarikh Pohon', and 'Butiran'. The first row has values 3, 3, Ravi Kumar, 03-12-2025, and three dots (...). The second row has values 11, 2, Mohd Hafiz, 19-01-2025, and three dots (...). A small blue square with the number '1' is centered below the table.

```

1 // Display the member details in loan list
2 SELECT tb_loan.l_loanApplicationID, tb_loan.l_memberNo,
3 tb_member.m_name,
4 DATE_FORMAT(tb_loan.l_applicationDate, '%d-%m-%Y') AS
5 formattedDate
6
7 FROM tb_loan
8 LEFT JOIN tb_member ON tb_loan.l_memberNo =
9 tb_member.m_memberNo
10
11 WHERE l_status = 1";
12

```

Loan Applicant Details

The screenshot shows a web-based application interface. At the top, there is a blue header bar with the logo 'MFI' on the left and a 'Log Keluar' button on the right. Below the header, a navigation menu includes links for 'Permohonan Anggota', 'Permohonan Pinjaman', 'Senarai Anggota', 'Senarai Peminjam', 'Transaksi', 'Kemaskini Polisi', 'Kemaskini Iklan', and 'Laporan'. The main content area is titled 'Maklumat Peminjam' and displays a table with various details about the loan application. The table has two columns: 'Maklumat Peribadi Pemohon' and '11'. The rows include: 'No. Aplikasi Pinjaman' (11), 'No. Anggota' (2), 'Nama Peminjam' (1004), 'Nama Anggota' (Mohd Hafiz), 'Jenis Pinjaman' (Karnival Musim Istimewa), 'Jumlah Pinjaman' (7000), 'Tempoh Pinjaman' (6), 'Ansuran Bulanan' (126.39), 'Akaun Bank' (78946632), 'Nama Bank' (BSN), and 'Gaji Kasar' (5800).

Gaji Bersih	5600
Tandatangan	
Pengesahan Majikan	Lihat
Tarikh Pohon	2025-01-19 12:57:06

Maklumat Penjamin 1	
No. Anggota	1
Nama Penjamin	Tan Mei Ling
No. Kad Pengenalan	050111-04-1234
No. PF	1
Tandatangan Penjamin 1	

Maklumat Penjamin 2	
No. Anggota	3
Nama Penjamin	Ravi Kumar
No. Kad Pengenalan	870401-06-0555
No. PF	1005
Tandatangan Penjamin 2	

Status Anggota

Pilih Status ▾

[Kembali](#) [Hantar](#)

```

1 // Display the loan applicant details
2 "SELECT l.* , m.m_name, m.m_pfNo, lt.lt_desc, b.lb_id, b.lb_desc
3   FROM tb_loan l
4     LEFT JOIN tb_member m ON l.l_memberNo = m.m_memberNo
5     LEFT JOIN tb_ltype lt ON l.l_loanType = lt.lt_lid
6     LEFT JOIN tb_lbanks b ON l.l_bankName = b.lb_id
7   WHERE l.l_loanApplicationID = '$lApplicationID'";
```

```

11 // Get the information of guarantor for loan application
12 "SELECT g.g_memberNo, g.g_signature FROM tb_guarantor g WHERE
13 g.g_loanApplicationID = '$lApplicationID' LIMIT 2"
14
15
16 // Fetch member details for guarantor 1
17 "SELECT m.m_name, m.m_ic, m.m_pfNo, m.m_memberNo FROM tb_member m
18 WHERE m.m_memberNo = '$guarantorID1'";
19
20
21
22 // Fetch member details for guarantor 2
23 "SELECT m.m_name, m.m_ic, m.m_pfNo, m.m_memberNo FROM tb_member m
24 WHERE m.m_memberNo = '$guarantorID2'";
25

```

View Loan Applicant List after approved

Senarai Peminjam

No. Pinjaman	No. Anggota	Nama Anggota	Jumlah Yang Dipinjam	Tarikh Lulus	Butiran
1	1	Tan Mei Ling	2000	2025-12-27 00:00:00	...
2	2	Mohd Hafiz	10000	2025-12-27 00:00:00	...
6	6	Aminah Saad	10000	2024-01-02 00:00:00	...
8	1	Tan Mei Ling	10000	2023-01-02 00:00:00	...
9	3	Ravi Kumar	50	2025-01-03 00:00:00	...
10	3	Ravi Kumar	50	2025-01-04 00:00:00	...

```

1 // Display the loan applicant list
2 "SELECT * FROM tb_loan
3     LEFT JOIN tb_member ON tb_loan.l_memberNo =
4         tb_member.m_memberNo
5     WHERE l_status = 3";
6
7

```

View Loan Applicant Details

Maklumat Pinjaman

Maklumat Pinjaman	
No. Aplikasi Pinjaman	1
Jenis Pinjaman	Al-Bai
Jumlah Pinjaman	2000
Tempoh Pinjaman	2
Ansuran Bulanan	90.33

Maklumat Peribadi Peminjam

Maklumat Peribadi Peminjam	
No. Anggota	1
No. PF	1
Nama Anggota	Tan Mei Ling
Akaun Bank	214748364
Nama Bank	Al Rajhi Bank Malaysia
Gaji Kasar	10000
Gaji Bersih	8000
Tarikh Pohon	2025-12-01 00:00:00
Tarikh Lulus	2025-12-27 00:00:00

Maklumat Penjamin 1

Maklumat Penjamin 1	
No. Anggota	3
Nama Penjamin	Ravi Kumar
No. Kad Pengenalan	870401-06-0555
No. PF	1005

Maklumat Penjamin 2

Maklumat Penjamin 2	
No. Anggota	3
Nama Penjamin	Ravi Kumar
No. Kad Pengenalan	870401-06-0555
No. PF	1005

[Kembali](#)

```

1 // Display the loan applicant's details
2 "SELECT tb_loan.* , tb_member.m_name , tb_member.m_pfNo ,
3 tb_ltype.lt_desc , tb_lbank.lb_desc
4
5     FROM tb_loan
6
7         LEFT JOIN tb_member ON tb_loan.l_memberNo =
8 tb_member.m_memberNo
9
10        LEFT JOIN tb_ltype ON tb_loan.l_loanType =
11 tb_ltype.lt_lid
12
13

```

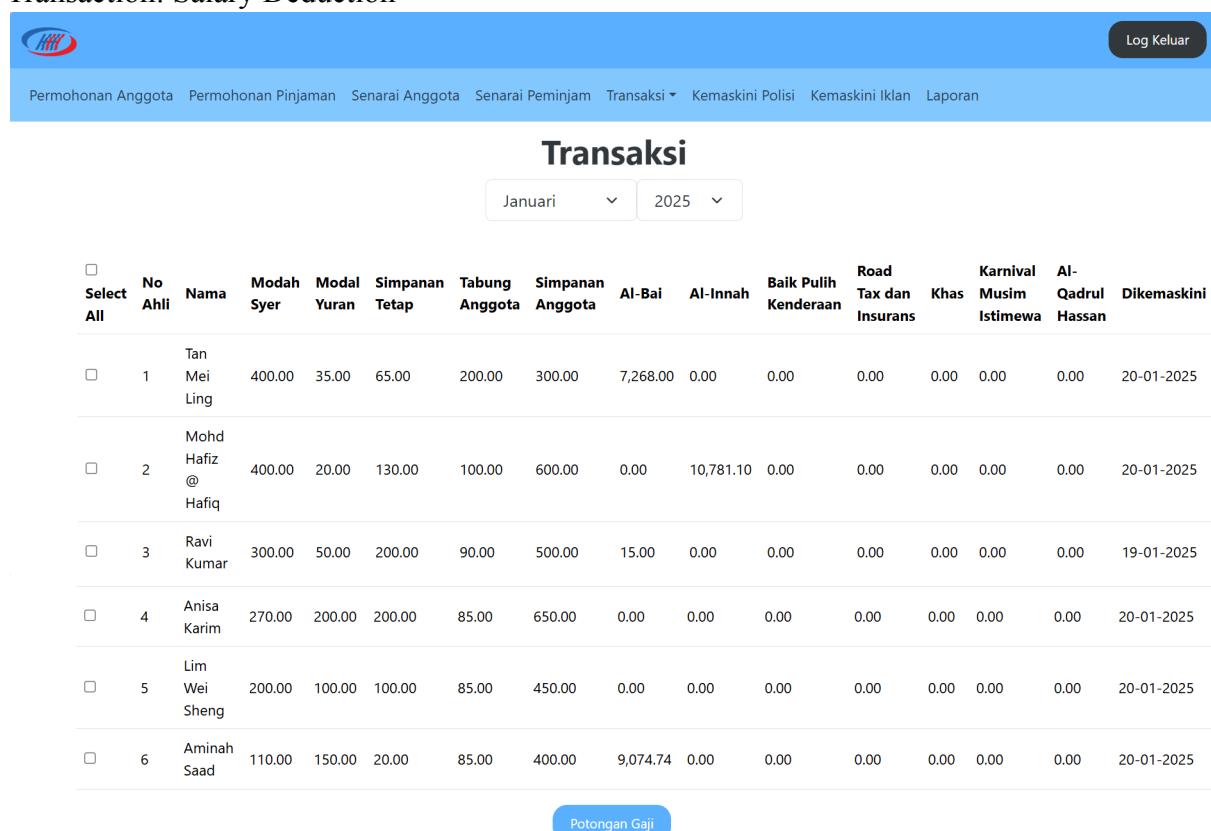
```

14         LEFT JOIN tb_lbank ON tb_loan.l_bankName =
15             tb_lbank.lb_id
16             WHERE tb_loan.l_loanApplicationID = '$loan_id'";
17
18
19 // Get the information of guarantor for loan application
20 "SELECT g.g_memberNo, g.g_signature FROM tb_guarantor g WHERE
21 g.g_loanApplicationID = '$loan_id' LIMIT 2";
22
23
24 // Fetch member details for guarantor 1
25 "SELECT m.m_name, m.m_ic, m.m_pfNo, m.m_memberNo FROM tb_member m
26 WHERE m.m_memberNo = '$guarantorID1'";
27
28
29 // Fetch member details for guarantor 2
30 "SELECT m.m_name, m.m_ic, m.m_pfNo, m.m_memberNo FROM tb_member m
31 WHERE m.m_memberNo = '$guarantorID2'";
32
33
34

```

Use Case: Manage System

Transaction: Salary Deduction



The screenshot shows a web application interface for managing transactions. At the top, there is a blue header bar with the logo of a bank or financial institution on the left and a 'Log Keluar' button on the right. Below the header, a navigation menu includes links for Permohonan Anggota, Permohonan Pinjaman, Senarai Anggota, Senarai Peminjam, Transaksi (with a dropdown arrow), Kemaskini Polisi, Kemaskini Iklan, and Laporan.

The main content area is titled 'Transaksi' (Transactions) in bold. Below the title, there are two dropdown menus for selecting the month ('Januari') and year ('2025').

A table lists six transactions, each with a checkbox in the first column:

	No Ahli	Nama	Modah Syer	Modal Yuran	Simpanan Tetap	Tabung Anggota	Simpanan Anggota	Al-Bai	Al-Innah	Baik Pulih Kenderaan	Road Tax dan Insurans	Khas	Karnival Musim Istimewa	Al-Qadrul Hassan	Dikemaskini	
<input type="checkbox"/>	1	Tan Mei Ling	400.00	35.00	65.00	200.00	300.00	7,268.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	20-01-2025
<input type="checkbox"/>	2	Mohd Hafiz @ Hafiq	400.00	20.00	130.00	100.00	600.00	0.00	10,781.10	0.00	0.00	0.00	0.00	0.00	0.00	20-01-2025
<input type="checkbox"/>	3	Ravi Kumar	300.00	50.00	200.00	90.00	500.00	15.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	19-01-2025
<input type="checkbox"/>	4	Anisa Karim	270.00	200.00	200.00	85.00	650.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	20-01-2025
<input type="checkbox"/>	5	Lim Wei Sheng	200.00	100.00	100.00	85.00	450.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	20-01-2025
<input type="checkbox"/>	6	Aminah Saad	110.00	150.00	20.00	85.00	400.00	9,074.74	0.00	0.00	0.00	0.00	0.00	0.00	0.00	20-01-2025

At the bottom center of the table area is a blue button labeled 'Potongan Gaji' (Salary Deduction).

```

1  -- Retrieve financial list of all members
2  SELECT tb_financial.*, tb_member.m_name
3      FROM tb_financial
4      INNER JOIN tb_member
5          ON tb_financial.f_memberNo=tb_member.m_memberNo;
6
7  -- Retrieve the sum of active loan by type of each member
8  SELECT l_memberNo, l_loanType, COALESCE(SUM(l_loanPayable), 0) AS
9  totalLoan
10     FROM tb_loan
11     WHERE l_status = 3
12     GROUP BY l_memberNo, l_loanType
13     ORDER BY l_memberNo, l_loanType;
14
15  -- Retrieve the month for selection
16  SELECT * FROM tb_rmonth;
17
18  -- Retrieve the loan types for display
19  SELECT * FROM tb_lttype
20      ORDER BY lt_lid;

```

Transaction: Salary Deduction Verification

The screenshot shows a web application interface for managing financial transactions. At the top, there is a navigation bar with links for 'Permohonan Anggota', 'Permohonan Pinjaman', 'Senarai Anggota', 'Senarai Peminjam', 'Transaksi', 'Kemaskini Polisi', 'Kemaskini Iklan', and 'Laporan'. On the right side of the header is a 'Log Keluar' button.

The main content area is titled 'Pengesahan Transaksi Potongan Gaji' (Salary Deduction Transaction Approval). It displays two tables, one for each member:

No Ahli	Nama	Perubahan Status				Tindakan
		Perkara	Status Semasa	Perubahan	Status Baharu	
1	Tan Mei Ling	Tabung Anggota	200.00	5.00	205.00	Amaran: Transaksi telah wujud untuk bulan ini!
		Simpanan Anggota	65.00	55.00	120.00	Keluarkan
Jumlah Potongan Gaji: RM 602.00						
2	Mohd Hafiz @ Hafiq	Perkara	Status Semasa	Perubahan	Status Baharu	Amaran: Transaksi telah wujud untuk bulan ini!
		Tabung Anggota	105.00	5.00	110.00	Keluarkan
Jumlah Potongan Gaji: RM 233.89						

3	Ravi Kumar	Perkara	Status Semasa	Perubahan	Status Baharu	Amaran: Transaksi telah wujud untuk bulan ini!		
		Modah Syer	355.00	45.00	400.00			
		Tabung Anggota	95.00	5.00	100.00			
		Simpanan Anggota	200.00	10.00	210.00			
Keluarkan								
ID Pinjaman Tangukkan Semasa Bayaran Tangukkan Baharu								
10 Al-Bai 10.00 5.00 5.00								
Jumlah Potongan Gaji: RM 65.00								
4	Anisa Karim	Perkara	Status Semasa	Perubahan	Status Baharu	Keluarkan		
		Modah Syer	270.00	55.00	325.00			
		Tabung Anggota	85.00	5.00	90.00			
		Jumlah Potongan Gaji: RM 60.00						
5	Lim Wei Sheng	Perkara	Status Semasa	Perubahan	Status Baharu	Keluarkan		
		Modah Syer	200.00	55.00	255.00			
		Tabung Anggota	85.00	5.00	90.00			
		Jumlah Potongan Gaji: RM 60.00						
6	Aminah Saad	Perkara	Status Semasa	Perubahan	Status Baharu	Keluarkan		
		Modah Syer	110.00	55.00	165.00			
		Tabung Anggota	85.00	5.00	90.00			
		ID Pinjaman Tangukkan Semasa Bayaran Tangukkan Baharu						
6 Al-Bai 9,074.74 312.18 8,762.56								
Jumlah Potongan Gaji: RM 372.18								
Hantar								

```

1 -- Retrieve policy information for calculation
2 SELECT p_minShareCapital, p_salaryDeductionForSaving,
3 p_salaryDeductionForMemberFund
4 FROM tb_policies
5 ORDER BY p_policyID DESC
6 LIMIT 1;
7
8 -- Retrieve the financial information by member
9 SELECT * FROM tb_financial WHERE f_memberNo = $memberNo;
10
11 -- Retrieve member information by member
12 SELECT * FROM tb_member WHERE m_memberNo = $memberNo;
13
14 -- Retrieve active loan information by member
15 SELECT tb_loan.*, tb_ltype.lt_desc
16   FROM tb_loan
17   JOIN tb_ltype ON tb_loan.l_loanType = tb_ltype.lt_lid
18   WHERE tb_loan.l_memberNo = $memberNo AND tb_loan.l_status = 3;
19
20 -- Check if the salary is deducted for the selected month
21 SELECT COUNT(t_transactionID) FROM tb_transaction
22   WHERE t_memberNo = $memberNo
23   AND t_month = $f_month

```

```

24    AND t_year = $f_year
25    AND t_method = 'Potongan Gaji';
26
27 -- Backend process after submission
28 -- Update the financial status after calculation
29 UPDATE tb_financial
30   SET f_shareCapital = f_shareCapital + '$shareCapitalChange',
31     f_feeCapital = f_feeCapital + '$feeCapitalChange',
32     f_fixedSaving = f_fixedSaving + '$fixedSavingChange',
33     f_memberSaving = f_memberSaving + '$memberSavingChange',
34     f_memberFund = f_memberFund + '$memberFundChange'
35 WHERE f_memberNo = '$memberNo';
36
37 -- Record the transaction to the transaction table
38 INSERT INTO tb_transaction(t_transactionType, t_method,
39 t_transactionAmt, t_month, t_year, t_desc, t_memberNo, t_adminID)
40   VALUES ('1', 'Potongan Gaji', '$difference', '$f_month', '$f_year',
41 'Potongan Gaji', '$memberNo', '$admin_id');
42
43 INSERT INTO tb_transaction(t_transactionType, t_method,
44 t_transactionAmt, t_month, t_year, t_desc, t_memberNo, t_adminID)
45   VALUES ('3', 'Potongan Gaji', '$difference', '$f_month', '$f_year',
46 'Potongan Gaji', '$memberNo', '$admin_id');
47
48 INSERT INTO tb_transaction(t_transactionType, t_method,
49 t_transactionAmt, t_month, t_year, t_desc, t_memberNo, t_adminID)
50   VALUES ('4', 'Potongan Gaji', '$difference', '$f_month', '$f_year',
51 'Potongan Gaji', '$memberNo', '$admin_id');
52
53 INSERT INTO tb_transaction(t_transactionType, t_method,
54 t_transactionAmt, t_month, t_year, t_desc, t_memberNo, t_adminID)
55   VALUES ('$transactionType', 'Potongan Gaji', $difference,
56 '$f_month', '$f_year', '$desc', '$memberNo', '$admin_id');
57
58 -- Update loan table
59 UPDATE tb_loan
60   SET l_loanPayable = $newLoanPayable,
61     l_status = $status
62 WHERE l_loanApplicationID = $loanID;

```

Transaction: Other transactions

Permohonan Anggota Permohonan Pinjaman Senarai Anggota Senarai Peminjam Transaksi ▾ Kemaskini Polisi Kemaskini Iklan Laporan

Transaksi Lain

No Anggota



Log Keluar

Permohonan Anggota Permohonan Pinjaman Senarai Anggota Senarai Peminjam Transaksi Kemaskini Polisi Kemaskini Iklan Laporan

Transaksi Lain

No Anggota

Maklumat Anggota

Name	Mohd Hafiz @ Hafiq
IC	880701-05-0444
No. Anggota	2
No. PF	1004

Maklumat Saham Ahli

	Status Semasa	Perubahan	Status Baru
Modal Syer	400.00	0.00	400.00
Modal Yuran	20.00	0.00	20.00
Simpanan Tetap	185.00	0.00	185.00
Simpanan Anggota	600.00	0.00	600.00
Tabung Anggota	105.00	0.00	105.00

Maklumat Pinjaman Ahli

ID	Pinjaman	Tangukkan Semasa	Ansuran Bulanan	Bayaran	Tangukkan Baharu
2	Al-Innah	10,607.21	173.89	0.00	10,607.21

Jumlah Bayaran: RM 0.00

Ulasan

Hantar

```

1 -- Retrieve the financial and member information by member
2 SELECT tb_financial.*, tb_member.m_name, tb_member.m_ic,
3 tb_member.m_pfNo
4   FROM tb_financial
5   INNER JOIN tb_member
6     ON tb_financial.f_memberNo=tb_member.m_memberNo
7     WHERE f_memberNo = '$memberNo';
8
9 -- Retrieve active loan information by member
10 SELECT tb_loan.*, tb_ltype.lt_desc
11   FROM tb_loan
12   JOIN tb_ltype ON tb_loan.l_loanType = tb_ltype.lt_lid
13     WHERE tb_loan.l_memberNo = $memberNo AND tb_loan.l_status = 3;
14

```

```

15  -- Backend process after submission
16  -- Record the transaction to the transaction table
17  INSERT INTO tb_transaction (t_transactionType, t_method,
18  t_transactionAmt, t_month, t_year, t_desc, t_memberNo, t_adminID)
19  VALUES ('$type', 'Transaksi Tambahan', '$changeAmount',
20  '$currentMonth', '$currentYear', '$desc', '$memberNo', '$admin_id');
21
22  -- Update loan table
23  UPDATE tb_loan
24  SET l_loanPayable = '$newLoanPayable',
25  l_status = '$status'
26  WHERE l_loanApplicationID = '$loanID';
27
28  -- Retrieve loan types to calculate transaction type
29  SELECT l_loanType from tb_loan
30  WHERE l_loanApplicationID = '$loanID';
31
32  -- Record the transaction to the transaction table
33  INSERT INTO tb_transaction (t_transactionType, t_method,
34  t_transactionAmt, t_month, t_year, t_desc, t_memberNo, t_adminID)
35  VALUES ('$tttype', 'Transaksi Tambahan', '$paymentAmount',
36  '$currentMonth', '$currentYear', '$desc', '$memberNo', '$admin_id');
37
38  -- Update the financial table
39  UPDATE tb_financial
40  SET f_shareCapital = f_shareCapital + '$shareCapitalChange',
41  f_feeCapital = f_feeCapital + '$feeCapitalChange',
42  f_fixedSaving = f_fixedSaving + '$fixedSavingChange',
43  f_memberSaving = f_memberSaving + '$memberSavingChange',
44  f_memberFund = f_memberFund + '$memberFundChange'
45  WHERE f_memberNo = '$memberNo';

```

Update Policies

Fee Masuk	RM
Modah Syer Minimum	RM400.00
Modal Yuran Minimum	RM35.00
Wang Deposit Anggota Minimum	RM50.00
Sumbangan Tabung Kebajikan Minimum	RM5.00
Simpanan Tetap Minimum	RM5.00
Lain-lain	RM0.00

Polisi Permohonan Pembiayaan

Kemaskini

Modal Syer Minimum Peminjam	RM 300.00
Kadar Keuntungan	4.20 %
Tempoh Ansuran Maksima	8 tahun
Jumlah Pembiayaan Maksima	RM 20,000.00

Jadual Pembayaran Balik Pembiayaan Skim Al-Bai / Al-Innah

Kadar Keuntungan		4.20 %								
Tempoh (Tahun)	1	2	3	4	5	6	7	8		
Tempoh (Bulan)	12	24	36	48	60	72	84	96		
Jumlah Pembiayaan				Ansuran Bulanan						
1,000.00	86.83	45.17	31.28	24.33	20.17	17.39	15.40	13.92		
2,000.00	173.67	90.33	62.56	48.67	40.33	34.78	30.81	27.83		
3,000.00	260.50	135.50	93.83	73.00	60.50	52.17	46.21	41.75		
4,000.00	347.33	180.67	125.11	97.33	80.67	69.56	61.62	55.67		
5,000.00	434.17	225.83	156.39	121.67	100.83	86.94	77.02	69.58		
6,000.00	521.00	271.00	187.67	146.00	121.00	104.33	92.43	83.50		
7,000.00	607.83	316.17	218.94	170.33	141.17	121.72	107.83	97.42		
8,000.00	694.67	361.33	250.22	194.67	161.33	139.11	123.24	111.33		
9,000.00	781.50	406.50	281.50	219.00	181.50	156.50	138.64	125.25		
10,000.00	868.33	451.67	312.78	243.33	201.67	173.89	154.05	139.17		
11,000.00	955.17	496.83	344.06	267.67	221.83	191.28	169.45	153.08		
12,000.00	1,042.00	542.00	375.33	292.00	242.00	208.67	184.86	167.00		
13,000.00	1,128.83	587.17	406.61	316.33	262.17	226.06	200.26	180.92		
14,000.00	1,215.67	632.33	437.89	340.67	282.33	243.44	215.67	194.83		
15,000.00	1,302.50	677.50	469.17	365.00	302.50	260.83	231.07	208.75		
16,000.00	1,389.33	722.67	500.44	389.33	322.67	278.22	246.48	222.67		
17,000.00	1,476.17	767.83	531.72	413.67	342.83	295.61	261.88	236.58		
18,000.00	1,563.00	813.00	563.00	438.00	363.00	313.00	277.29	250.50		
19,000.00	1,649.83	858.17	594.28	462.33	383.17	330.39	292.69	264.42		
20,000.00	1,736.67	903.33	625.56	486.67	403.33	347.78	308.10	278.33		

Polisi Potongan Gaji

Kemaskini

Simpanan Tetap	RM 55.00
Sumbangan Tabung Kebajikan	RM 5.00

```

1 -- Retrieve the latest policy with newest ID
2 SELECT * FROM tb_policies
3 ORDER BY p_policyID DESC
4 LIMIT 1;

```

Update Policies: Basic Policies for Member Application

Kemaskini Polisi Asas Pemohonan Anggota

Fee Masuk

RM 35 .00

Modah Syer Minimum

RM 400 .00

Modal Yuran Minimum

RM 35 .00

Wang Deposit Anggota Minimum

RM 50 .00

Sumbangan Tabung Kebajikan Minimum

RM 5 .00

Simpanan Tetap Minimum

RM 5 .00

Lain-lain

RM 0 .00

[Kembali](#) [Kemaskini](#)

```
1 -- Retrieve the latest policy with newest ID
2 SELECT * FROM tb_policies
3     ORDER BY p_policyID DESC
4     LIMIT 1;
5
6 -- Backend process after submission
7 -- Insert new record
8 INSERT INTO tb_policies (
9     p_memberRegFee, p_minShareCapital, p_minFeeCapital,
10    p_minFixedSaving, p_minMemberFund, p_minMemberSaving, p_minOtherFees,
11    p_minShareCapitalForLoan, p_profitRate, p_maxInstallmentPeriod,
12    p_maxFinancingAmt,
13    p_salaryDeductionForSaving, p_salaryDeductionForMemberFund,
14    p_adminID)
15 VALUES (
16    '$f_memberRegFee', '$f_minShareCapital', '$f_minFeeCapital',
17    '$f_minFixedSaving', '$f_minMemberFund', '$f_minMemberSaving',
18    '$f_minOtherFees',
19    '$d_minShareCapitalForLoan', '$d_profitRate',
20    '$d_maxInstallmentPeriod', '$d_maxFinancingAmt',
```

```

21 '$d_salaryDeductionForSaving', '$d_salaryDeductionForMemberFund',
22 '$admin_id');
23

```

Update Policies: Financing Policies

Modal Syer Minimum Peminjam

RM	300	.00
----	-----	-----

Kadar Keuntungan

4.2	%
-----	---

Tempoh Ansuran Maksima

8	tahun
---	-------

Jumlah Pembiayaan Maksima

RM	10000	.00
----	-------	-----

Kadar Keuntungan								
Tempoh (Tahun)	1	2	3	4	5	6	7	8
Tempoh (Bulan)	12	24	36	48	60	72	84	96
Jumlah Pembiayaan	Ansuran Bulanan							
1000.00	86.83	45.17	31.28	24.33	20.17	17.39	15.40	13.92
2000.00	173.67	90.33	62.56	48.67	40.33	34.78	30.81	27.83
3000.00	260.50	135.50	93.83	73.00	60.50	52.17	46.21	41.75
4000.00	347.33	180.67	125.11	97.33	80.67	69.56	61.62	55.67
5000.00	434.17	225.83	156.39	121.67	100.83	86.94	77.02	69.58
6000.00	521.00	271.00	187.67	146.00	121.00	104.33	92.43	83.50
7000.00	607.83	316.17	218.94	170.33	141.17	121.72	107.83	97.42
8000.00	694.67	361.33	250.22	194.67	161.33	139.11	123.24	111.33
9000.00	781.50	406.50	281.50	219.00	181.50	156.50	138.64	125.25
10000.00	868.33	451.67	312.78	243.33	201.67	173.89	154.05	139.17

Kembali	Kemaskini
---------	-----------

```

1  -- Retrieve the latest policy with newest ID
2  SELECT * FROM tb_policies
3      ORDER BY p_policyID DESC
4      LIMIT 1;
5
6  -- Backend process after submission
7  -- Insert new record
8  INSERT INTO tb_policies (
9      p_memberRegFee, p_minShareCapital, p_minFeeCapital,
10     p_minFixedSaving, p_minMemberFund, p_minMemberSaving, p_minOtherFees,
11     p_minShareCapitalForLoan, p_profitRate, p_maxInstallmentPeriod,
12     p_maxFinancingAmt,
13     p_salaryDeductionForSaving, p_salaryDeductionForMemberFund,
14     p_adminID)
15 VALUES (
16     '$d_memberRegFee', '$d_minShareCapital', '$d_minFeeCapital',
17     '$d_minFixedSaving', '$d_minMemberFund', '$d_minMemberSaving',
18     '$d_minOtherFees',
19     '$f_minShareCapitalForLoan', '$f_profitRate',
20     '$f_maxInstallmentPeriod', '$f_maxFinancingAmt',
21     '$d_salaryDeductionForSaving', '$d_salaryDeductionForMemberFund',
22     '$admin_id');
23

```

Update Policies: Salary Deduction Policies

Simpanan Tetap

RM 55 .00

Sumbangan Tabung Kebajikan

RM 5 .00

Kembali Kemaskini

```

1  -- Retrieve the latest policy with newest ID
2  SELECT * FROM tb_policies
3      ORDER BY p_policyID DESC
4      LIMIT 1;
5
6  -- Backend process after submission
7  -- Insert new record
8  INSERT INTO tb_policies (
9      p_memberRegFee, p_minShareCapital, p_minFeeCapital,
10     p_minFixedSaving, p_minMemberFund, p_minMemberSaving, p_minOtherFees,
11     p_minShareCapitalForLoan, p_profitRate, p_maxInstallmentPeriod,
12     p_maxFinancingAmt,
13     p_salaryDeductionForSaving, p_salaryDeductionForMemberFund,
14     p_adminID)

```

```

15 VALUES (
16   '$d_memberRegFee', '$d_minShareCapital', '$d_minFeeCapital',
17   '$d_minFixedSaving', '$d_minMemberFund', '$d_minMemberSaving',
18   '$d_minOtherFees',
19   '$d_minShareCapitalForLoan', '$d_profitRate',
20   '$d_maxInstallmentPeriod', '$d_maxFinancingAmt',
21   '$f_salaryDeductionForSaving', '$f_salaryDeductionForMemberFund',
22   '$admin_id');
23

```

Update Banner

Kemaskini Iklan

Iklan Semasa

Kemaskini

**Perlu Pinjaman?
Dapatkan
Kelulusan
Cepat!**

**Kadar Faedah Rendah
Terma Pembayaran
Fleksibel
Permohonan Cepat &
Mudah**

Muat Naik Iklan Baru

Nama Iklan Baru

Muat Naik Iklan Baharu

Choose File No file chosen

Muat Naik

```

1 -- Retrieve banners that are active
2 SELECT * FROM tb_banner WHERE b_status = 1;
3
4 -- Backend process after upload
5 INSERT INTO tb_banner (b_banner, b_name, b_status, b_adminID)
6 VALUES ('$bannerPath', '$bannerName', '$b_status', '$admin_id')

```

Update banner to be displayed

Nama	Iklan	Status	Tindakan
Default		Aktif	
Sky		Tidak aktif	
Keluar		Tidak aktif	
Awas		Tidak aktif	

1

[Kembali](#)

```

1 -- Retrieve banners regardless of status with pagination
2 SELECT * FROM tb_banner
3     ORDER BY b_bannerID ASC
4     LIMIT $start_from, $records_per_page;
5
6 -- Count the number of banner for pagination calculation
7 SELECT COUNT(*) FROM tb_banner;
8
9 -- Backend processes
10 -- Retrieve banner to check existence before deleting
11 SELECT b_banner FROM tb_banner
12     WHERE b_bannerID = $bannerID;
13
14 -- Delete banner
15 DELETE FROM tb_banner
16     WHERE b_bannerID = $bannerID;
17
18 -- Update banner status
19 UPDATE tb_banner
20     SET b_status = $status
21     WHERE b_bannerID = $bannerID;

```

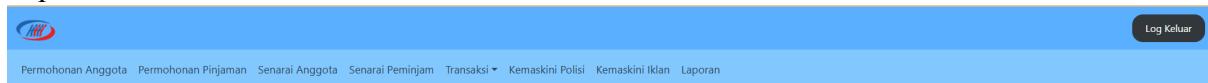
```

22
23 -- Check the number of active banners
24 SELECT COUNT(*) FROM tb_banner
25 WHERE b_status = 1;

```

Use Case: Generate Report

Report Dashboard



Bulan Tahun
 -- Pilih Bulan -- Contoh: 2024 Tapis

Laporan Kewangan

Sila pilih bulan dan/atau tahun untuk melihat laporan.

Laporan Kewangan Disember 2024

1. Ringkasan Eksekutif
 Laporan ini memberikan analisis terperinci mengenai prestasi kewangan syarikat untuk tempoh Disember 2024, termasuk metrik utama seperti permohonan ahli, permohonan pinjaman, rekod transaksi dan kesihatan kewangan keseluruhan.

2. Gambaran Keseluruhan Permohonan Ahli
 Jumlah Permohonan Ahli Baru: 4
 Permohonan Mengikut Status:
 • Diluluskan: 2
 • Sedang Diproses: 1
 • Ditolak: 1

3. Gambaran Keseluruhan Permohonan Pinjaman
 Jumlah Permohonan Pinjaman Baru: 4
 Permohonan Mengikut Status:
 • Diluluskan: 2
 • Sedang Diproses: 1
 • Ditolak: 1

Permohonan Pinjaman Mengikut Jenis:

Permohonan Anggota Permohonan Pinjaman Senarai Anggota Senarai Peminjam Transaksi Kemaskini Polisi Kemaskini Iklan Laporan

• [Orangku](#)

Permohonan Pinjaman Mengikut Jenis:

- Al-Bai: 2
- Al-Innah: 0
- Baik Pulih Kenderaan: 1
- Road Tax dan Insurans: 1
- Khas: 0
- Karnival Musim Istimewa: 0
- Al-Qadru Hassan: 0

4. Prestasi Transaksi

Jumlah Transaksi: 3

Jumlah Amaun Transaksi: RM 450.00

5. Maklumat Polisi

- Yuran Pendaftaran Anggota: 35
- Modal Syer Minimum: 300
- Yuran Modal Minimum: 0
- Yuran Tetap Minimum: 50
- Simpanan Tetap Minimum: 5
- Tabung Kebajikan Minimum: 50
- Jumlah Pembiayaan Maksimum: 40000
- Kadar Keuntungan: 5

6. Prestasi Kewangan

Jumlah Pendapatan untuk Tempoh: RM 450.00

Jumlah Perbelanjaan untuk Tempoh: RM 12,000.00

Keuntungan Bersih untuk Tempoh: RM -11,550.00

```

1 // Retrieve data from tb_rmonth for display month
2 if ($bulan && $tahun) {
3     $month_query = mysqli_query
4         ($con, "SELECT rm_desc
5             FROM tb_rmonth
6             WHERE rm_id = '$bulan'");
7     if ($month_query && mysqli_num_rows($month_query) > 0) {
8         $month_name = mysqli_fetch_assoc($month_query) ['rm_desc'];
9         $period = $month_name . " " . $tahun;
10    }
11 } elseif ($bulan) {
12     $month_query = mysqli_query
13         ($con, "SELECT rm_desc
14             FROM tb_rmonth
15             WHERE rm_id = '$bulan'");
16     if ($month_query && mysqli_num_rows($month_query) > 0) {
17         $month_name = mysqli_fetch_assoc($month_query) ['rm_desc'];
18         $period = $month_name;
19     }
20 } elseif ($tahun) {
21     $period = "Tahun " . $tahun;
22 }
23
24 // Retrieve data from tb_member
25 $member_sql = "SELECT COUNT(*) AS member_count
26                 FROM tb_member $where_clause";
27 $member_result = mysqli_query($con, $member_sql);
28
29 // Retrieve data from tb_member group by status
30 $status_sql = "SELECT m_status, COUNT(*) AS status_count
31                 FROM tb_member
32                 $where_clause
33                 GROUP BY m_status";
34 $status_result = mysqli_query($con, $status_sql);
35
36 // Retrieve data from tb_loan

```

```

37 $loan_where = str_replace('m_applicationDate', 'l_applicationDate',
38 $where_clause);
39 $loan_sql = "SELECT COUNT(*) AS loan_count
40     FROM tb_loan $loan_where";
41 $loan_result = mysqli_query($con, $loan_sql);
42
43 // Retrieve data from tb_loan group by status
44 $loan_status_sql = "SELECT l_status, COUNT(*) AS loan_status_count,
45     SUM(CASE WHEN l_status = 3 THEN l_appliedLoan
46 ELSE 0 END) AS total_approved_loans
47     FROM tb_loan $loan_where
48     GROUP BY l_status";
49 $loan_status_result = mysqli_query($con, $loan_status_sql);
50
51 // Retrieve data from tb_loan group by type
52 $loan_type_sql = "SELECT l_loanType, COUNT(*) AS loan_type_count
53     FROM tb_loan
54     $loan_where
55     GROUP BY l_loanType";
56 $loan_type_result = mysqli_query($con, $loan_type_sql);
57
58 // Retrieve data from tb_transaction
59 $transaction_where = str_replace('m_applicationDate',
60 't_transactionDate', $where_clause);
61 $transactions_sql = "SELECT COUNT(*) AS transaction_count,
62     SUM(t_transactionAmt) AS transaction_total
63     FROM tb_transaction $transaction_where";
64 $transactions_result = mysqli_query($con, $transactions_sql);
65
66 // Retrieve data from tb_policies
67 $policies_where = str_replace('m_applicationDate', 'p_dateUpdated',
68 $where_clause);
69 $policies_sql = "SELECT * FROM tb_policies $policies_where";
70 $policies_result = mysqli_query($con, $policies_sql);

```

View in PDF

The screenshot shows a PDF document with the following details:

- LAPORAN KEWANGAN**
- Tempoh: Disember 2024
- 1. Ringkasan Eksekutif**
 - Laporan ini memberikan analisis terperinci mengenai prestasi kewangan syarikat untuk tempoh Disember 2024, termasuk metrik utama seperti permohonan ahli, permohonan pinjaman, rekod transaksi dan kesihatan kewangan keseluruhan.
- 2. Gambaran Keseluruhan Permohonan Ahli**

Status	Jumlah
Diluluskan	2
Sedang Diproses	1
Ditolak	1
JUMLAH PERMOHONAN	4
- 3. Gambaran Keseluruhan Permohonan Pinjaman**

Status	Jumlah
Diluluskan	2
Sedang Diproses	1
Ditolak	1
JUMLAH PERMOHONAN	4

Jumlah Amaun Pinjaman: RM 37,000.00

Muka 1/3

LAPORAN KEWANGAN

4. Prestasi Transaksi

Perkara	Jumlah
Bilangan Transaksi	3
JUMLAH AMAUN	RM 450.00

5. Prestasi Kewangan

Perkara	Jumlah
Pendapatan	RM 450.00
Perbelanjaan	RM 37,000.00
KEUNTUNGAN BERSIH	RM -36,550.00

6. Maklumat Polisi

Perkara	Nilai
Yuran Pendaftaran Anggota	RM 35.00
Modal Syer Minimum	RM 300.00
Yuran Modal Minimum	RM 0.00
Yuran Tetap Minimum	RM 50.00
Simpanan Tetap Minimum	RM 5.00
Tabung Kebajikan Minimum	RM 50.00
Jumlah Pembayaran Maksimum	RM 40,000.00
Kadar Keuntungan	5%

7. Kesimpulan

Berdasarkan analisis untuk tempoh Disember 2024, laporan ini merumuskan bahawa:

1. Permohonan Keahlian:
 - Kadar kelulusan keahlian adalah 50.0%
 - Jumlah 4 permohonan baharu telah diterima
2. Permohonan Pinjaman:
 - Kadar kelulusan pinjaman adalah 50.0%
 - Jumlah nilai pinjaman yang diluluskan: RM 37,000.00
3. Prestasi Kewangan:
 - Syarikat mencatatkan kerugian bersih sebanyak RM 36,550.00

Maka 2/3

LAPORAN KEWANGAN

4. Cadangan:

- Meningkatkan proses kelulusan keahlian
- Mengkaji semula kriteria kelayakan pinjaman
- Mengkaji semula strategi pengurusan kewangan
- Memantau prestasi kewangan secara berterusan
- Mengelakkan polisi yang sedia ada untuk kostabilan operasi

```

1 // Retrieve data from tb_rmonth for display month
2 if ($bulan && $tahun) {
3     $month_query = mysqli_query
4         ($con, "SELECT rm_desc
5             FROM tb_rmonth
6             WHERE rm_id = '$bulan'");
7 } elseif ($bulan) {
8     $month_query = mysqli_query
9         ($con, "SELECT rm_desc
10            FROM tb_rmonth
11            WHERE rm_id = '$bulan'");
12 } elseif ($tahun) {
13     $period = "Tahun " . $tahun;
14 } else {
15     $period = "Semua Tempoh";
16 }
17
18 // Retrieve data from tb_member
19 $member_sql = "SELECT
20     COUNT(*) as total_count,
21     SUM(CASE WHEN m_status = 1 THEN 1 ELSE 0 END) as
pending_count,
22     SUM(CASE WHEN m_status = 2 THEN 1 ELSE 0 END) as
rejected_count,
23     SUM(CASE WHEN m_status = 3 THEN 1 ELSE 0 END) as
approved_count
24     FROM tb_member" . createWhereClause($bulan, $tahun,
25 'm_applicationDate');
26
27     error_log("Member SQL: " . $member_sql);
28
29
30     $member_result = mysqli_query($con, $member_sql);
31     $member_data = mysqli_fetch_assoc($member_result);
32
33
34 // Retrieve data from tb_loan
35 $loan_sql = "SELECT
36     COUNT(*) as total_count,
37     SUM(CASE WHEN l_status = 1 THEN 1 ELSE 0 END) as
pending_count,
38     SUM(CASE WHEN l_status = 2 THEN 1 ELSE 0 END) as
rejected_count,
39     SUM(CASE WHEN l_status = 3 THEN 1 ELSE 0 END) as
approved_count,
40     SUM(l_appliedLoan) as total_amount
41     FROM tb_loan" . createWhereClause($bulan, $tahun,
42 'l_applicationDate');
43
44
45     $loan_result = mysqli_query($con, $loan_sql);
46     $loan_data = mysqli_fetch_assoc($loan_result);
47
48
49
50 $loan_type_sql = "SELECT
51     SUM(CASE WHEN l_loanType = 1 THEN 1 ELSE 0 END) as alBai,
52     SUM(CASE WHEN l_loanType = 2 THEN 1 ELSE 0 END) as alInnah,
53     SUM(CASE WHEN l_loanType = 3 THEN 1 ELSE 0 END) as
baikPulihKenderaan,
54
55

```

```

56     SUM(CASE WHEN l_loanType = 4 THEN 1 ELSE 0 END) as roadTax,
57     SUM(CASE WHEN l_loanType = 5 THEN 1 ELSE 0 END) as khas,
58     SUM(CASE WHEN l_loanType = 6 THEN 1 ELSE 0 END) as karnival,
59     SUM(CASE WHEN l_loanType = 7 THEN 1 ELSE 0 END) as alQadrul
60     FROM tb_loan" . createWhereClause($bulan, $tahun,
61 'l_applicationDate');

62
63     $loan_type_result = mysqli_query($con, $loan_type_sql);
64     $loan_type_data = mysqli_fetch_assoc($loan_type_result);
65
66 // Retrieve data from tb_transaction
67 $transaction_sql = "SELECT
68     COUNT(*) as transaction_count,
69     COALESCE(SUM(t_transactionAmt), 0) as transaction_total
70     FROM tb_transaction" . createWhereClause($bulan, $tahun,
71 't_transactionDate');

72
73     $transaction_result = mysqli_query($con, $transaction_sql);
74     $transaction_data = mysqli_fetch_assoc($transaction_result);
75
76 // Retrieve data from tb_policies
77 $policies_sql = "SELECT * FROM tb_policies";
78     if ($bulan && $tahun) {
79         $policies_sql .= " WHERE MONTH(p_dateUpdated) = '$bulan' AND
80 YEAR(p_dateUpdated) = '$tahun'";
81     }
82     $policies_sql .= " ORDER BY p_dateUpdated DESC LIMIT 1";
83
84     $policies_result = mysqli_query($con, $policies_sql);
85     $policy_data = mysqli_fetch_assoc($policies_result);

```

Save report retrieval log

The screenshot shows a web application interface. At the top, there is a navigation bar with links: Permohonan Anggota, Permohonan Pinjaman, Senarai Anggota, and Senarai Peminjam. On the right side of the header, there is a 'Log Keluar' button. Below the header, there are two dropdown menus: 'Bulan' set to 'Disember' and 'Tahun' set to '2024'. A 'Tapis' button is located next to the year dropdown. In the center, there is a title 'Laporan Kewangan Disember 2024'. To the right of the title are two buttons: 'Simpan Log' and 'Lihat dalam PDF'. A modal dialog box is displayed in the center, with the text 'localhost says' at the top and 'Log berjaya disimpan' below it. An 'OK' button is at the bottom right of the modal. The main content area below the title contains sections for executive summary, client loan applications, and loan application details.

```

1 // Insert data into tb_reportretrievallog
2 $sql = "INSERT INTO tb_reportretrievallog (r_retrievalDate, r_month,
3 r_year, r_adminID)

```

```

4      VALUES (CURRENT_TIMESTAMP(), ?, ?, ?)";
5
6 $stmt = $con->prepare($sql);
7 if (!$stmt) {
8     throw new Exception("Prepare failed: " . $con->error);
9 }
10
11 $stmt->bind_param("sss", $month, $year, $adminNo);
12 if (!$stmt->execute()) {
13     throw new Exception("Execute failed: " . $stmt->error);
14 }
15
16 if ($stmt->affected_rows > 0) {
17     echo json_encode(['status' => 'success']);
18 } else {
19     throw new Exception("No rows inserted");
20 }

```

4.2 SQL Statement

4.2.1 Data Definition Language (DDL)

tb_member

```

1 -- Table structure for table `tb_member`
2 --
3
4
5 CREATE TABLE `tb_member` (
6     `m_memberApplicationID` int(11) NOT NULL,
7     `m_name` varchar(50) NOT NULL,
8     `m_ic` varchar(14) NOT NULL,
9     `m_email` varchar(255) NOT NULL,
10    `m_gender` int(11) NOT NULL,
11    `m_religion` int(11) NOT NULL,
12    `m_race` int(11) NOT NULL,
13    `m_maritalStatus` int(11) NOT NULL,
14    `m_homeAddress` varchar(50) NOT NULL,
15    `m_homePostcode` int(11) NOT NULL,
16    `m_homeCity` varchar(50) NOT NULL,
17    `m_homeState` int(11) NOT NULL,
18    `m_memberNo` int(11) DEFAULT NULL,
19    `m_pfNo` int(11) DEFAULT NULL,
20    `m_position` varchar(50) NOT NULL,
21    `m_positionGrade` varchar(11) NOT NULL,
22    `m_officeAddress` varchar(50) NOT NULL,
23
24
25
26
27
28
29
30

```

```

31   `m_officePostcode` int(11) NOT NULL,
32   `m_officeCity` varchar(50) NOT NULL,
33   `m_officeState` int(11) NOT NULL,
34   `m_phoneNumber` varchar(13) NOT NULL,
35   `m_homeNumber` varchar(13) DEFAULT NULL,
36   `m_monthlySalary` int(11) NOT NULL,
37   `m_feeMasuk` double NOT NULL,
38   `m_modalSyer` double NOT NULL,
39   `m_modalYuran` double NOT NULL,
40   `m_deposit` double NOT NULL,
41   `m_alAbrar` double NOT NULL,
42   `m_simpananTetap` double NOT NULL COMMENT 'Sumbangan Tabung
43 Tabung Kebajikan ',
44   `m_feeLain` double NOT NULL,
45   `m_status` int(11) NOT NULL,
46   `m_applicationDate` datetime DEFAULT NULL,
47   `m_approvalDate` timestamp NULL DEFAULT NULL,
48   `m_adminID` int(11) DEFAULT NULL COMMENT 'Admin who approve
49 the application'
50 )
51 -- Indexes for table `tb_member`
52 --
53
54 ALTER TABLE `tb_member`
55   ADD PRIMARY KEY (`m_memberApplicationID`),
56   ADD UNIQUE KEY `m_memberApplicationID`(
57     `m_memberApplicationID`),
58   ADD UNIQUE KEY `m_memberNo`(`m_memberNo`),
59   ADD UNIQUE KEY `m_memberNo_2`(`m_memberNo`),
60   ADD KEY `m_gender`(
61     `m_gender`, `m_religion`, `m_race`, `m_maritalStatus`, `m_status`),
62   ADD KEY `m_religion`(`m_religion`),
63   ADD KEY `m_race`(`m_race`),
64   ADD KEY `m_maritalStatus`(`m_maritalStatus`),
65   ADD KEY `m_status`(`m_status`),
66   ADD KEY `m_adminID`(`m_adminID`),
67   ADD KEY `m_homeState`(`m_homeState`, `m_officeState`),
68   ADD KEY `tb_member_ibfk_7`(`m_officeState`);

69 --
70 -- AUTO_INCREMENT for table `tb_member`
71 --

```

```

88 ALTER TABLE `tb_member`
89     MODIFY `m_memberApplicationID` int(11) NOT NULL
90     AUTO_INCREMENT;
91 --
92 --
93 -- Constraints for table `tb_member`
94 --
95 
96 ALTER TABLE `tb_member`
97     ADD CONSTRAINT `tb_member_ibfk_1` FOREIGN KEY (`m_gender`)
98 REFERENCES `tb_ugender` (`ug_gid`),
99     ADD CONSTRAINT `tb_member_ibfk_2` FOREIGN KEY (`m_religion`)
100 REFERENCES `tb ureligion` (`ua_rid`),
101     ADD CONSTRAINT `tb_member_ibfk_3` FOREIGN KEY (`m_race`)
102 REFERENCES `tb_urace` (`ur_rid`),
103     ADD CONSTRAINT `tb_member_ibfk_4` FOREIGN KEY
104     (`m_maritalStatus`) REFERENCES `tb_umaritalstatus` (`um_mid`),
105     ADD CONSTRAINT `tb_member_ibfk_5` FOREIGN KEY (`m_status`)
106 REFERENCES `tb_status` (`s_sid`),
107     ADD CONSTRAINT `tb_member_ibfk_6` FOREIGN KEY (`m_homeState`)
108 REFERENCES `tb_homestate` (`st_id`),
109     ADD CONSTRAINT `tb_member_ibfk_7` FOREIGN KEY
110     (`m_officeState`) REFERENCES `tb_officestate` (`st_id`);

111
112
113
114
115
116
117
118
119

```

tb_ltype

```

1 -- Table structure for table `tb_ltype`
2 --
3 
4 CREATE TABLE `tb_ltype` (
5     `lt_lid` int(11) NOT NULL,
6     `lt_desc` varchar(30) NOT NULL
7 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
8 --
9 --
10 -- Dumping data for table `tb_ltype`
11 --
12 
13 INSERT INTO `tb_ltype` (`lt_lid`, `lt_desc`) VALUES
14 (1, 'Al-Bai'),
15 (2, 'Al-Innah'),
16 (3, 'Baik Pulih Kenderaan'),
17 (4, 'Road Tax dan Insurans'),
18 (5, 'Khas'),
19 (6, 'Karnival Musim Istimewa'),
20 (7, 'Al-Qadrul Hassan');

```

```

21
22 -----
23 --
24 -- Indexes for table `tb_ltype`
25 --
26 ALTER TABLE `tb_ltype`
27   ADD PRIMARY KEY (`lt_lid`);

```

tb lbank

```

1 --
2 -- Table structure for table `tb_lbank`
3 --
4
5 CREATE TABLE `tb_lbank` (
6   `lb_id` int(11) NOT NULL,
7   `lb_desc` varchar(30) NOT NULL
8 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
9
10 --
11 --
12 -- Dumping data for table `tb_lbank`
13 --
14
15 INSERT INTO `tb_lbank`(`lb_id`, `lb_desc`) VALUES
16 (1, 'Affin Bank'),
17 (2, 'Agrobank'),
18 (3, 'Al Rajhi Bank Malaysia'),
19 (4, 'Alliance Bank'),
20 (5, 'AmBank'),
21 (6, 'Bank Islam'),
22 (7, 'Bank Muamalat'),
23 (8, 'Bank Rakyat'),
24 (9, 'BSN'),
25 (10, 'CIMB'),
26 (11, 'Citibank Malaysia'),
27 (12, 'Co-op Bank Pertama'),
28 (13, 'Hong Leong Bank'),
29 (14, 'HSBC Malaysia'),
30 (15, 'Maybank'),
31 (16, 'MBSB Bank'),
32 (17, 'OCBC Malaysia'),
33 (18, 'Public Bank'),
34 (19, 'RHB'),
35 (20, 'Standard Chartered Malaysia'),
36 (21, 'UOB Malaysia'),
37 (22, 'Tiada');

41
42 -----
43 --
44 -- Indexes for table `tb_lbank`
45 --

```

```

46 ALTER TABLE `tb_lbank`
47   ADD PRIMARY KEY (`lb_id`);
48
49

```

tb loan

```

1 -- Table structure for table `tb_loan`
2 --
3
4 CREATE TABLE `tb_loan` (
5   `l_loanApplicationID` int(11) NOT NULL,
6   `l_memberNo` int(11) NOT NULL,
7   `l_loanType` int(11) NOT NULL,
8   `l_appliedLoan` double NOT NULL,
9   `l_loanPeriod` int(11) NOT NULL COMMENT '(Years)',
10  `l_monthlyInstalment` double NOT NULL COMMENT '\r\n',
11  `l_loanPayable` double NOT NULL,
12  `l_bankAccountNo` int(11) NOT NULL,
13  `l_bankName` int(11) NOT NULL,
14  `l_monthlyGrossSalary` double NOT NULL,
15  `l_monthlyNetSalary` double NOT NULL,
16  `l_signature` varchar(50) NOT NULL,
17  `l_file` varchar(50) NOT NULL COMMENT '(Pengesahan Majikan)',
18  `l_status` int(11) NOT NULL,
19  `l_applicationDate` datetime DEFAULT NULL,
20  `l_approvalDate` timestamp NULL DEFAULT NULL,
21  `l_adminID` int(11) DEFAULT NULL COMMENT 'Admin who approved the
22 application'
23 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
24
25 --
26
27 --
28
29 -- Indexes for table `tb_loan`
30 --
31
32 ALTER TABLE `tb_loan`
33   ADD PRIMARY KEY (`l_loanApplicationID`),
34   ADD KEY `l_loanType` (`l_loanType`),
35   ADD KEY `l_adminID` (`l_adminID`),
36   ADD KEY `l_status` (`l_status`),
37   ADD KEY `l_memberNo` (`l_memberNo`),
38   ADD KEY `l_bankName` (`l_bankName`);
39
40 --
41
42 --
43 -- AUTO_INCREMENT for table `tb_loan`
44 --
45 ALTER TABLE `tb_loan`
46   MODIFY `l_loanApplicationID` int(11) NOT NULL AUTO_INCREMENT,
47   AUTO_INCREMENT=28;
48
49 --
50

```

```

51 -- Constraints for table `tb_loan`
52 --
53 ALTER TABLE `tb_loan`
54   ADD CONSTRAINT `tb_loan_ibfk_1` FOREIGN KEY (`l_adminID`) REFERENCES
55   `tb_admin` (`a_adminID`),
56   ADD CONSTRAINT `tb_loan_ibfk_1` FOREIGN KEY (`l_loanType`)
57   REFERENCES `tb_ltype` (`lt_lid`),
58   ADD CONSTRAINT `tb_loan_ibfk_2` FOREIGN KEY (`l_status`) REFERENCES
59   `tb_status` (`s_sid`),
60   ADD CONSTRAINT `tb_loan_ibfk_3` FOREIGN KEY (`l_memberNo`)
61   REFERENCES `tb_member` (`m_memberNo`),
62   ADD CONSTRAINT `tb_loan_ibfk_4` FOREIGN KEY (`l_bankName`)
63   REFERENCES `tb_lbank` (`lb_id`);
64
65
66
67

```

tb_guarantor

```

1 --
2 -- Table structure for table `tb_guarantor`
3 --
4 --
5 CREATE TABLE `tb_guarantor` (
6   `g_guarantorID` int(11) NOT NULL,
7   `g_loanApplicationID` int(11) NOT NULL,
8   `g_memberNo` int(11) NOT NULL,
9   `g_signature` varchar(50) NOT NULL
10 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
11
12 --
13 --
14 -- Indexes for table `tb_guarantor`
15 --
16 ALTER TABLE `tb_guarantor`
17   ADD PRIMARY KEY (`g_guarantorID`),
18   ADD KEY `g_loanApplicationID` (`g_loanApplicationID`),
19   ADD KEY `g_memberNo` (`g_memberNo`);
20
21 --
22 --
23 -- AUTO_INCREMENT for table `tb_guarantor`
24 --
25 ALTER TABLE `tb_guarantor`
26   MODIFY `g_guarantorID` int(11) NOT NULL AUTO_INCREMENT,
27   AUTO_INCREMENT=35;
28
29 --
30 --
31 -- Constraints for table `tb_guarantor`
32 --
33 ALTER TABLE `tb_guarantor`
34   ADD CONSTRAINT `tb_guarantor_ibfk_1` FOREIGN KEY
35   (`g_loanApplicationID`) REFERENCES `tb_loan` (`l_loanApplicationID`),

```

```

36     ADD CONSTRAINT `tb_guarantor_ibfk_2` FOREIGN KEY (`g_memberNo`)
37 REFERENCES `tb_member` (`m_memberNo`);
38
39 --
40
41
42

```

tb_rmonth

```

1  --
2  -- Table structure for table `tb_rmonth`
3  --
4
5  CREATE TABLE `tb_rmonth` (
6      `rm_id` int(11) NOT NULL,
7      `rm_desc` varchar(10) NOT NULL
8  ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
9
10 --
11 --
12 -- Dumping data for table `tb_rmonth`
13 --
14
15 INSERT INTO `tb_rmonth` (`rm_id`, `rm_desc`) VALUES
16 (1, 'Januari'),
17 (2, 'Februari'),
18 (3, 'Mac'),
19 (4, 'April'),
20 (5, 'Mei'),
21 (6, 'Jun'),
22 (7, 'Julai'),
23 (8, 'Ogos'),
24 (9, 'September'),
25 (10, 'Oktober'),
26 (11, 'November'),
27 (12, 'Disember');
28
29
30 -----
31 -- Indexes for table `tb_rmonth`
32 --
33
34 ALTER TABLE `tb_rmonth`
35     ADD PRIMARY KEY (`rm_id`);
36
37 --
38
39

```

tb_reportretrievallog

```

1  --
2  -- Table structure for table `tb_reportretrievallog`
3  --
4

```

```

5 CREATE TABLE `tb_reportretrievallog` (
6   `r_retrievalID` int(11) NOT NULL,
7   `r_retrievalDate` date NOT NULL,
8   `r_month` int(11) NOT NULL,
9   `r_year` int(11) NOT NULL,
10  `r_adminID` int(11) NOT NULL
11 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
12
13 --
14 -- Indexes for table `tb_reportretrievallog`
15 --
16 ALTER TABLE `tb_reportretrievallog`
17   ADD PRIMARY KEY (`r_retrievalID`),
18   ADD KEY `r_adminID` (`r_adminID`),
19   ADD KEY `tb_reportretrievallog_ibfk_2` (`r_month`);
20
21 --
22 -- AUTO_INCREMENT for table `tb_reportretrievallog`
23 --
24 ALTER TABLE `tb_reportretrievallog`
25   MODIFY `r_retrievalID` int(11) NOT NULL AUTO_INCREMENT,
26 AUTO_INCREMENT=5;
27
28 --
29 -- Constraints for table `tb_reportretrievallog`
30 --
31 ALTER TABLE `tb_reportretrievallog`
32   ADD CONSTRAINT `tb_reportretrievallog_ibfk_1` FOREIGN KEY
33   (`r_adminID`) REFERENCES `tb_admin` (`a_adminID`),
34   ADD CONSTRAINT `tb_reportretrievallog_ibfk_2` FOREIGN KEY
35   (`r_month`) REFERENCES `tb_rmonth` (`rm_id`);
36
37 --
38
39
40
41

```

4.2.2 Data Manipulation Language (DML)

Use Case: Log In

```

1 //Retrieves data from tb_user for user availability by user id
2
3 $sql = "SELECT * FROM tb_user WHERE u_id = ?";
4 $stmt = mysqli_prepare($con, $sql);
5 mysqli_stmt_bind_param($stmt, 'i', $fname);
6 mysqli_stmt_execute($stmt);
7 $result = mysqli_stmt_get_result($stmt);
8 $user = mysqli_fetch_assoc($result);
9 $matches = false;
10
11 if ($user) {

```

```

12     $matches = password_verify($fpwd, $user['u_pwd']);
13
14 //Retrieves data from tb_member for matching memberId and email
15
16 $sql = "SELECT m_memberNo, m_email
17     FROM tb_member
18     WHERE m_memberNo = ? AND m_email = ?";
19 $stmt = mysqli_prepare($con, $sql);
20 mysqli_stmt_bind_param($stmt, 'is', $userID, $email);
21 mysqli_stmt_execute($stmt);
22 $result = mysqli_stmt_get_result($stmt);
23
24 if (mysqli_num_rows($result) > 0) {
25 //generate a unique token that last 10 min
26     $token = bin2hex(random_bytes(16));
27     $expiryTime = date("Y-m-d H:i:s", strtotime("+10 minutes",
28 time()));
29
30
31 //Set reset_token and expiry time in tb_user by userID
32
33     $sql = "UPDATE tb_user SET reset_token = ?, token_expiry = ?
34 WHERE u_id = ?";
35     $stmt = mysqli_prepare($con, $sql);
36     mysqli_stmt_bind_param($stmt, 'ssi', $token, $expiryTime,
37 $userID);
38     mysqli_stmt_execute($stmt);
39
40 //check for matching token and expiry status in tb_user by reset token
41
42     $sql = "SELECT * FROM tb_user WHERE reset_token = ?";
43     $stmt = mysqli_prepare($con, $sql);
44     mysqli_stmt_bind_param($stmt, 's', $token);
45     mysqli_stmt_execute($stmt);
46     $result = mysqli_stmt_get_result($stmt);
47     $user = mysqli_fetch_assoc($result);
48
49 //reset password by reset token
50     $sql = "UPDATE tb_user SET u_pwd = ?, reset_token = NULL,
51 token_expiry = NULL WHERE reset_token = ?";
52     $stmt = mysqli_prepare($con, $sql);
53     mysqli_stmt_bind_param($stmt, 'ss', $hashedPassword, $token);
54     mysqli_stmt_execute($stmt);
55
56
57
58

```

Use Case: Apply for Membership

```

1 //finds for similar ic and email in tb_member to check for
2 duplicated entries
3 $fuic = $cleanPost['fuic'];
4 $sql = "SELECT * FROM tb_member WHERE m_ic = ?";
5 $binds = [$fuic];
6 $result = query($sql, $binds);
7

```

```

8     $row = mysqli_fetch_assoc($result);
9
10    if ($row) {
11        //display error message
12    }
13
14    $femail = $cleanPost['femail'];
15    $sql_email = "SELECT * FROM tb_member WHERE m_email = ?";
16    $binds_email = [$femail];
17    $result_email = query($sql_email, $binds_email);
18
19
20    if ($row_email = mysqli_fetch_assoc($result_email)) {
21        display error message
22    }
23
24    //retrieves a single latest minimum value of fees from tb_policies
25
26    $minValues = [];
27    $query = "
28        SELECT
29            p_memberRegFee AS ffee,
30            p_minShareCapital AS fmodal,
31            p_minFeeCapital AS fyuran,
32            p_minFixedSaving AS fanggota,
33            p_minMemberFund AS fabrar,
34            p_minMemberSaving AS ftetap,
35            p_minOtherFees AS fother
36        FROM tb_policies
37        ORDER BY p_policyID DESC
38        LIMIT 1";
39
40    $result = mysqli_query($con, $query);
41    if ($result && $row = mysqli_fetch_assoc($result)) {
42        $minValues = $row;
43    }
44
45    //registration (insert into database)
46    include('..\db_connect.php');
47    $fdate = date('Y-m-d H:i:s');
48
49    // Start transaction
50    $con->begin_transaction();
51
52    // Insert into tb_member
53    $stmt = $con->prepare("
54        INSERT INTO tb_member (
55            m_name, m_ic, m_email, m_gender, m_religion, m_race,
56            m_maritalStatus, m_homeAddress, m_homeCity, m_homeState,
57            m_homePostcode,
58            m_position, m_positionGrade, m_pfNo, m_officeAddress,
59            m_officeCity, m_officeState, m_officePostcode,
60            m_phoneNumber, m_homeNumber, m_monthlySalary, m_feeMasuk,
61            m_modalSyer, m_modalYuran, m_deposit,
62            m_alAbrar, m_simpananTetap, m_feeLain, m_applicationDate,
63            m_status
64
```

```

65      ) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?,
66      ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)
67      ");
68
69     $stmt->bind_param(
70         "sssssssssssssssssssssssssssssssssssssssi",
71         $funame,
72         $fuic,
73         $femail,
74         $fgender,
75         $freligion,
76         $frace,
77         $fmariage,
78         $fhomeaddress,
79         $fcity,
80         $fstate,
81         $fhomezip,
82         $fposition,
83         $fgrade,
84         $fpfno,
85         $fofficeaddress,
86         $fofficecity,
87         $fofficestate,
88         $fofficezip,
89         $ftelno,
90         $fhomeno,
91         $fsalary,
92         $fee,
93         $fmodal,
94         $fyuran,
95         $fanggota,
96         $fabrar,
97         $ftetap,
98         $fother,
99         $fdate,
100        $fstatus
101    );
102
103    if ($stmt->execute()) {
104        // Get the last inserted member ID
105        $member_id = $con->insert_id;
106
107        // Prepare heir insert statement
108        $stmt_heir = $con->prepare("
109            INSERT INTO tb_heir (h_memberApplicationID, h_name, h_ic,
110            h_relationWithMember)
111            VALUES (?, ?, ?, ?)
112        ");
113
114        if ($stmt_heir) {
115            // Insert heirs
116            foreach ($heirs as $heir) {
117                $stmt_heir->bind_param("isss", $member_id,
118                $heir['name'], $heir['ic'], $heir['relation']);
119                if (!$stmt_heir->execute()) {
120                    // Rollback on error
121                    $con->rollback();

```

```

122                     echo "<script>alert('Error inserting heir data: " .
123 . $stmt_heir->error . "');</script>";
124                     exit();
125                 }
126             }
127
128         // Commit the transaction
129         $con->commit();
130
131
132 //display information
133 if (isset($_GET['member_id']) && !empty($_GET['member_id'])) {
134     $member_id = intval($_GET['member_id']); // Sanitize input
135
136     // Retrieve member details from tb_member by memberID for display
137     purpose
138     $stmt = $con->prepare("SELECT * FROM tb_member
139     LEFT JOIN tb_ugender ON tb_member.m_gender = tb_ugender.ug_gid
140     LEFT JOIN tb_ureligion ON tb_member.m_religion =
141     tb_ureligion.ua_rid
142     LEFT JOIN tb_urace ON tb_member.m_race = tb_urace.ur_rid
143     LEFT JOIN tb_homestate ON tb_member.m_homeState =
144     tb_homestate.st_id
145     LEFT JOIN tb_officestate ON tb_member.m_officeState =
146     tb_officestate.st_id
147     WHERE m_memberApplicationID = ?");
148     $stmt->bind_param("i", $member_id);
149     $stmt->execute();
150     $member_result = $stmt->get_result();
151     $member_data = $member_result->fetch_assoc();
152
153     // Retrieve heir details from tb_heir by memberID for display
154     purpose
155
156     $stmt = $con->prepare("SELECT * FROM tb_heir
157     LEFT JOIN tb_hrelation ON tb_heir.h_relationWithMember =
158     tb_hrelation.hr_rid
159     WHERE h_memberApplicationID = ?");
160     $stmt->bind_param("i", $member_id);
161     $stmt->execute();
162     $heirs_result = $stmt->get_result();
163     $heirs_data = $heirs_result->fetch_all(MYSQLI_ASSOC);
164
165     $stmt->close();
166 }
167

```

Use Case: View Loan Application Status

1	//Retrieve data from tb_loan for dashboard loan
2	

```

3 $sql = "SELECT l_loanType, l_appliedLoan, l_loanPeriod,
4 l_monthlyInstalment, l_loanPayable, l_status, l_applicationDate
5         FROM tb_loan
6         WHERE l_memberNo= $memberNo";
7 $result = mysqli_query($con, $sql);
8
9     while ($row = mysqli_fetch_assoc($result)) {
10         $selected_jenis_pembayaran_ID =
11 htmlspecialchars($row['l_loanType']);
12         $selected_amaunDipohon = htmlspecialchars($row['l_appliedLoan']);
13         $selected_tempohPembayaran = htmlspecialchars($row['l_loanPeriod']);
14         $selected_ansuranBulanan =
15 htmlspecialchars($row['l_monthlyInstalment']);
16         $selected_tunggakan = htmlspecialchars($row['l_loanPayable']);
17         $status_loan_ID = htmlspecialchars($row['l_status']);
18         $application_date = htmlspecialchars($row['l_applicationDate']);
19
20 //Retrieve data from tb_ltype for loan type
21 if ($selected_jenis_pembayaran_ID) {
22     $sql_desc = "SELECT lt_desc
23             FROM tb_ltype
24             WHERE lt_lid = $selected_jenis_pembayaran_ID";
25     $desc_result = mysqli_query($con, $sql_desc);
26
27     if ($desc_row = mysqli_fetch_assoc($desc_result)) {
28         $selected_jenis_pembayaran = $desc_row['lt_desc'];
29     }
30 }
31 //Retrieve data from tb_status for loan status
32 if ($status_loan_ID) {
33     $sql_status = "SELECT s_desc
34             FROM tb_status
35             WHERE s_sid = $status_loan_ID";
36     $status_result = mysqli_query($con, $sql_status);
37
38     if ($status_row = mysqli_fetch_assoc($status_result)) {
39         $status_loan = $status_row['s_desc'];
40     }
41 }
42 }
43

```

Use Case: Apply Loan

```

1 // Retrieve data from tb_policies and tb_financial to make sure
2 // applicant able to apply loan and check for loan details does not
3 // exceed in tb_policies
4
5 $sql = "SELECT * FROM tb_policies
6             ORDER BY p_policyID DESC
7             LIMIT 1;";
8     $result = mysqli_query($con, $sql);
9     $policy = mysqli_fetch_assoc($result);
10
11 $sql = "SELECT f_shareCapital

```

```

12      FROM tb_financial
13      WHERE f_memberNo = $memberNo";
14  $result = mysqli_query($con, $sql);
15
16  if ($row = mysqli_fetch_assoc($result)) {
17      $shareCapital = htmlspecialchars($row['f_shareCapital']);
18  }
19
20 // Insert data to tb_loan
21 $sql = "INSERT INTO tb_loan (l_memberNo, l_loanType, l_appliedLoan,
22 l_loanPeriod, l_monthlyInstalment, l_loanPayable, l_bankAccountNo,
23 l_bankName, l_monthlyGrossSalary, l_monthlyNetSalary, l_signature,
24 l_status, l_applicationDate)
25     VALUES ('$memberNo', '$jenis_pembayaran', '$amaunDipohon',
26 '$tempohPembayaran', '$ansuranBulanan', '$tunggakan', '$bankAcc',
27 '$namaBank', '$gajiKasar', '$gajiBersih', '$fileSign', 1,
28 CURRENT_TIMESTAMP())";
29
30 //Retrieve data from tb_member for checking details
31 if ($memberNo !== null) {
32     $sql = "SELECT m_name, m_ic, m_email, m_gender, m_religion, m_race,
33 m_maritalStatus, m_pfNo, m_monthlySalary, m_homeAddress,
34 m_homePostcode, m_homeCity, m_homeState, m_positionGrade, m_position,
35 m_officeAddress, m_officePostcode, m_officeCity, m_officeState,
36 m_homeNumber, m_phoneNumber
37     FROM tb_member
38     WHERE m_memberNo = '$memberNo'";
39 $result = mysqli_query($con, $sql);
40
41 if ($row = mysqli_fetch_assoc($result)) {
42     $memberName = htmlspecialchars($row['m_name']);
43     $memberIC = htmlspecialchars($row['m_ic']);
44     $memberEmail = htmlspecialchars($row['m_email']);
45     $memberGender = $row['m_gender'];
46     $memberReligion = ($row['m_religion']);
47     $memberRace = ($row['m_race']);
48     $memberMaritalStatus = ($row['m_maritalStatus']);
49     $memberpfNo = htmlspecialchars($row['m_pfNo']);
50     $memberHomeAdd = htmlspecialchars($row['m_homeAddress']);
51     $memberHomePostcode = htmlspecialchars($row['m_homePostcode']);
52     $memberHomeCity = htmlspecialchars($row['m_homeCity']);
53     $memberHomeState = htmlspecialchars($row['m_homeState']);
54     $memberPositionGrade = htmlspecialchars($row['m_positionGrade']);
55     $memberPosition = htmlspecialchars($row['m_position']);
56     $memberOfficeAdd = htmlspecialchars($row['m_officeAddress']);
57     $memberOfficePostcode =
58     htmlspecialchars($row['m_officePostcode']);
59     $memberOfficeCity = htmlspecialchars($row['m_officeCity']);
60     $memberOfficeState = htmlspecialchars($row['m_officeState']);
61     $memberHomeTele = htmlspecialchars($row['m_homeNumber']);
62     $memberPhoneTele = htmlspecialchars($row['m_phoneNumber']);
63     $memberMonthlySalary = htmlspecialchars($row['m_monthlySalary']);
64
65 }
66 }
67
68 // Modify data in tb_member

```

```

69 if (!empty($memberNo)) {
70     $sql = "UPDATE tb_member
71         SET m_name = '$nama',
72             m_email = '$email',
73             m_gender = '$selectedGender',
74             m_religion = '$selectedReligion',
75             m_race = '$selectedRace',
76             m_maritalStatus = '$selectedMaritalRace',
77             m_monthlySalary = '$selectedMonthlySalary',
78
79             m_homeAddress = '$selectedHomeAdd',
80             m_homePostcode = '$selectedHomePostcode',
81             m_homeCity = '$selectedHomeCity',
82             m_homeState = '$selectedHomeState',
83
84             m_position = '$selectedPosition',
85             m_positionGrade = '$selectedPositionGrade',
86
87             m_officeAddress = '$selectedOfficeAdd',
88             m_officePostcode = '$selectedOfficePostcode',
89             m_officeCity = '$selectedOfficeCity',
90             m_officeState = '$selectedOfficeState',
91
92             m_phoneNumber = '$selectedNoTeleBim',
93             m_homeNumber = '$selectedNoTelePej'
94     WHERE m_memberNo = '$memberNo'";
95
96 } else {
97     echo "Error: Missing member number.";
98 }
99
100 // Retrieve data from tb_member
101 if ($memberNo !== null) {
102     $sql = "SELECT m_name, m_ic
103             FROM tb_member
104             WHERE m_memberNo = '$memberNo'";
105     $result = mysqli_query($con, $sql);
106 }
107 if ($row = mysqli_fetch_assoc($result)) {
108     $memberName = htmlspecialchars($row['m_name']);
109     $memberIC = htmlspecialchars($row['m_ic']);
110 }
111
112
113 //Retrieve data from tb_member for taking guarantor 1 details by "No.
114 //anggota"
115 if(isset($_POST['anggotaPenjamin1'])) {$memberNo =
116 mysqli_real_escape_string($con, $_POST['anggotaPenjamin1']);
117
118     $sql = "SELECT m_memberNo, m_name, m_ic, m_pfNo
119             FROM tb_member
120             WHERE m_memberNo = ?";
121
122     $stmt = mysqli_prepare($con, $sql);
123     mysqli_stmt_bind_param($stmt, "s", $memberNo);
124     mysqli_stmt_execute($stmt);
125     $result = mysqli_stmt_get_result($stmt);

```

```

126
127     if(mysqli_num_rows($result) > 0) {
128         $member = mysqli_fetch_assoc($result);
129         echo json_encode([
130             'status' => 'success',
131             'penjamin1' => [
132                 'm_name' => $member['m_name'],
133                 'm_ic' => $member['m_ic'],
134                 'm_pfNo' => $member['m_pfNo']
135             ]
136         ]);
137     } else {
138         echo json_encode([
139             'status' => 'error',
140             'message' => 'Tiada maklumat anggota dijumpai'
141         ]);
142     }
143     mysqli_stmt_close($stmt);
144 }
145
146 //Retrieve data from tb_member for taking guarantor 2 details by "No.
147 //anggota"
148 if(isset($_POST['anggotaPenjamin2'])) {$memberNo =
149 mysqli_real_escape_string($con, $_POST['anggotaPenjamin2']);
150
151     $sql = "SELECT m_memberNo, m_name, m_ic, m_pfNo
152         FROM tb_member
153         WHERE m_memberNo = ?";
154
155     $stmt = mysqli_prepare($con, $sql);
156     mysqli_stmt_bind_param($stmt, "s", $memberNo);
157     mysqli_stmt_execute($stmt);
158     $result = mysqli_stmt_get_result($stmt);
159
160     if(mysqli_num_rows($result) > 0) {
161         $member = mysqli_fetch_assoc($result);
162         echo json_encode([
163             'status' => 'success',
164             'penjamin2' => [
165                 'm_name' => $member['m_name'],
166                 'm_ic' => $member['m_ic'],
167                 'm_pfNo' => $member['m_pfNo']
168             ]
169         ]);
170     } else {
171         echo json_encode([
172             'status' => 'error',
173             'message' => 'Tiada maklumat anggota dijumpai'
174         ]);
175     }
176     mysqli_stmt_close($stmt);
177 }
178
179 //Insert data into tb_guarantor
180 $sql1 = "INSERT INTO tb_guarantor (g_loanApplicationID, g_memberNo,
181 g_signature)
182

```

```

183     VALUES ('$loanApplicationID', '$memberNo1',
184 '$fileSignPenjamin1');");
185
186 $sql2 = "INSERT INTO tb_guarantor (g_loanApplicationID, g_memberNo,
187 g_signature)
188     VALUES ('$loanApplicationID', '$memberNo2',
189 '$fileSignPenjamin2')";
190
191 //Retrieve data from tb_loan for checking details
192 $sql = "SELECT l_monthlyGrossSalary, l_monthlyNetSalary
193     FROM tb_loan
194     WHERE l_loanApplicationID = $loanApplicationID";
195 $result = mysqli_query($con, $sql);
196
197 if ($row = mysqli_fetch_assoc($result)) {
198     $gajiKasar = htmlspecialchars($row['l_monthlyGrossSalary']);
199     $gajiBersih = htmlspecialchars($row['l_monthlyNetSalary']);
200 }
201
202 // Retrieve data from tb_loan for display loan details
203 $sql = "SELECT l_loanType, l_appliedLoan, l_loanPeriod,
204 l_monthlyInstalment, l_bankAccountNo, l_bankName,
205 l_monthlyGrossSalary, l_monthlyNetSalary, l_signature
206     FROM tb_loan
207     WHERE l_loanApplicationID = $loanApplicationID";
208 $result = mysqli_query($con, $sql);
209
210 if ($row = mysqli_fetch_assoc($result)) {
211     $selected_jenis_pembiayaan_ID =
212 htmlspecialchars($row['l_loanType']);
213     $selected_amaunDipohon = htmlspecialchars($row['l_appliedLoan']);
214     $selected_tempohPembiayaan =
215 htmlspecialchars($row['l_loanPeriod']);
216     $selected_ansuranBulanan =
217 htmlspecialchars($row['l_monthlyInstalment']);
218     $selected_namaBank_ID = htmlspecialchars($row['l_bankName']);
219     $selected_bankAcc = htmlspecialchars($row['l_bankAccountNo']);
220     $selected_gajiKasar =
221 htmlspecialchars($row['l_monthlyGrossSalary']);
222     $selected_gajiBersih =
223 htmlspecialchars($row['l_monthlyNetSalary']);
224     $selected_signature = htmlspecialchars($row['l_signature']);
225 }
226
227 // Retrieve data from tb_ltype for display loan name
228 if ($selected_jenis_pembiayaan_ID) {
229     $sql = "SELECT lt_desc
230         FROM tb_ltype
231         WHERE lt_lid = $selected_jenis_pembiayaan_ID";
232     $result = mysqli_query($con, $sql);
233     if ($row = mysqli_fetch_assoc($result)) {
234         $selected_jenis_pembiayaan = $row['lt_desc'];
235     }
236 }
237
238 // Retrieve data from tb_lbank for display name bank
239 if ($selected_namaBank_ID) {

```

```

240     $sql = "SELECT lb_desc
241         FROM tb_lbank
242             WHERE lb_id = $selected_namaBank_ID";
243     $result = mysqli_query($con, $sql);
244     if ($row = mysqli_fetch_assoc($result)) {
245         $selected_namaBank = $row['lb_desc'];
246     }
247 }
248
249 // Retrieve data from tb_member for display member details
250 if ($memberNo !== null) {
251     $sql = "SELECT m_name, m_ic, m_email, m_gender, m_religion, m_race,
252 m_maritalStatus, m_pfNo, m_monthlySalary, m_homeAddress,
253 m_homePostcode, m_homeCity, m_homeState, m_positionGrade, m_position,
254 m_officeAddress, m_officePostcode, m_officeCity, m_officeState,
255 m_homeNumber, m_phoneNumber
256         FROM tb_member
257             WHERE m_memberNo = '$memberNo'";
258     $result = mysqli_query($con, $sql);
259
260     if ($row = mysqli_fetch_assoc($result)) {
261         $memberName = htmlspecialchars($row['m_name']);
262         $memberIC = htmlspecialchars($row['m_ic']);
263         $memberEmail = htmlspecialchars($row['m_email']);
264         $memberGender_ID = $row['m_gender'];
265         $memberReligion_ID = ($row['m_religion']);
266         $memberRace_ID = ($row['m_race']);
267         $memberMaritalStatus_ID = ($row['m_maritalStatus']);
268         $memberpfNo = htmlspecialchars($row['m_pfNo']);
269         $memberHomeAdd = htmlspecialchars($row['m_homeAddress']);
270         $memberHomePostcode = htmlspecialchars($row['m_homePostcode']);
271         $memberHomeCity = htmlspecialchars($row['m_homeCity']);
272         $memberHomeState_ID = ($row['m_homeState']);
273         $memberPositionGrade = htmlspecialchars($row['m_positionGrade']);
274         $memberPosition = htmlspecialchars($row['m_position']);
275         $memberOfficeAdd = htmlspecialchars($row['m_officeAddress']);
276         $memberOfficePostcode =
277             htmlspecialchars($row['m_officePostcode']);
278         $memberOfficeCity = htmlspecialchars($row['m_officeCity']);
279         $memberOfficeState_ID = ($row['m_officeState']);
280         $memberHomeTele = htmlspecialchars($row['m_homeNumber']);
281         $memberPhoneTele = htmlspecialchars($row['m_phoneNumber']);
282         $memberMonthlySalary = htmlspecialchars($row['m_monthlySalary']);
283     }
284 }
285
286 // Retrieve data from tb_ugender for display gender
287 if ($memberGender_ID) {
288     $sql = "SELECT ug_desc
289         FROM tb_ugender
290             WHERE ug_gid = $memberGender_ID";
291     $result = mysqli_query($con, $sql);
292     if ($row = mysqli_fetch_assoc($result)) {
293         $memberGender = $row['ug_desc'];
294     }
295 }
296

```

```

297 // Retrieve data from tb_ureligion for display religion
298 if ($memberReligion_ID) {
299     $sql = "SELECT ua_desc
300           FROM tb_ureligion
301             WHERE ua_rid = $memberReligion_ID";
302     $result = mysqli_query($con, $sql);
303     if ($row = mysqli_fetch_assoc($result)) {
304         $memberReligion = $row['ua_desc'];
305     }
306 }
307
308 // Retrieve data from tb_urace for display race
309 if ($memberRace_ID) {
310     $sql = "SELECT ur_desc
311           FROM tb_urace
312             WHERE ur_rid = $memberRace_ID";
313     $result = mysqli_query($con, $sql);
314     if ($row = mysqli_fetch_assoc($result)) {
315         $memberRace = $row['ur_desc'];
316     }
317 }
318
319 // Retrieve data from tb_umaritalstatus for display marital status
320 if ($memberMaritalStatus_ID) {
321     $sql = "SELECT um_desc
322           FROM tb_umaritalstatus
323             WHERE um_mid = $memberMaritalStatus_ID";
324     $result = mysqli_query($con, $sql);
325     if ($row = mysqli_fetch_assoc($result)) {
326         $memberMaritalStatus = $row['um_desc'];
327     }
328 }
329
330 // Retrieve data from tb_homestate for display home state
331 if ($memberHomeState_ID) {
332     $sql = "SELECT st_desc
333           FROM tb_homestate
334             WHERE st_id = $memberHomeState_ID";
335     $result = mysqli_query($con, $sql);
336     if ($row = mysqli_fetch_assoc($result)) {
337         $memberHomeState = $row['st_desc'];
338     }
339 }
340
341 // Retrieve data from tb_officestate for display office state
342 if ($memberOfficeState_ID) {
343     $sql = "SELECT st_desc
344           FROM tb_officestate
345             WHERE st_id = $memberOfficeState_ID";
346     $result = mysqli_query($con, $sql);
347     if ($row = mysqli_fetch_assoc($result)) {
348         $memberOfficeState = $row['st_desc'];
349     }
350 }
351
352 // Retrieve data from tb_guarantor for display guarantor 1 details
353

```

```

354 $sql = "SELECT g.g_memberNo, g.g_signature, m.m_name, m.m_ic,
355 m.m_pfNo
356     FROM tb_guarantor g
357     LEFT JOIN tb_member m ON g.g_memberNo = m.m_memberNo
358     WHERE g.g_guarantorID = '$guarantorID1'";
359 $result = mysqli_query($con, $sql);
360
361 if ($result && $row = mysqli_fetch_assoc($result)) {
362     $anggotaPenjamin1 = $row['g_memberNo'];
363     $signaturePenjamin1 = $row['g_signature'];
364     $namaPenjamin1 = $row['m_name'];
365     $icPenjamin1 = $row['m_ic'];
366     $pfPenjamin1 = $row['m_pfNo'];
367 }
368
369 // Retrieve data from tb_guarantor for display guarantor 2 details
370 $sql2 = "SELECT g.g_memberNo, g.g_signature, m.m_name, m.m_ic,
371 m.m_pfNo
372     FROM tb_guarantor g
373     LEFT JOIN tb_member m ON g.g_memberNo = m.m_memberNo
374     WHERE g.g_guarantorID = '$guarantorID2'";
375 $result2 = mysqli_query($con, $sql2);
376
377 if ($result2 && $row2 = mysqli_fetch_assoc($result2)) {
378     $anggotaPenjamin2 = $row2['g_memberNo'];
379     $signaturePenjamin2 = $row2['g_signature'];
380     $namaPenjamin2 = $row2['m_name'];
381     $icPenjamin2 = $row2['m_ic'];
382     $pfPenjamin2 = $row2['m_pfNo'];
383 }
384
385 // Retrieve data from tb_policies to check for loan details does not
386 // exceed in tb_policies
387 $sql = "
388     SELECT * FROM tb_policies
389     ORDER BY p_policyID DESC
390     LIMIT 1;";
391 $result = mysqli_query($con, $sql);
392 $policy = mysqli_fetch_assoc($result);
393
394 // Modify data in tb_loan
395 if (!empty($loanApplicationID)) {
396     $sql = "UPDATE tb_loan
397             SET l_loanType = '$loanType',
398                 l_appliedLoan = '$samaunDipohon',
399                 l_loanPeriod = '$tempohPembayaran',
400                 l_monthlyInstalment = '$ansuranBulanan',
401                 l_loanPayable = '$tunggakan',
402                 l_bankName = '$namaBank',
403                 l_bankAccountNo = '$bankAcc',
404                 l_monthlyGrossSalary = '$gajiKasar',
405                 l_monthlyNetSalary = '$gajiBersih',
406                 l_signature = '$fileSign'
407
408             WHERE l_loanApplicationID = '$loanApplicationID'";
409
410     echo "<br>SQL Query: " . $sql . "<br>";

```

```

411
412     if (mysqli_query($con, $sql)) {
413         echo "Update successful!<br>";
414         header('Location: semakan_butiran.php?status=success');
415         exit();
416     } else {
417         echo "Error updating record: " . mysqli_error($con);
418     }
419 } else {
420     echo "Error: Missing member number.";
421 }
422
423 //Retrieve data from tb_member for checking details
424 if ($memberNo !== null) {
425     $sql = "SELECT m_name, m_ic, m_email, m_gender, m_religion, m_race,
426 m_maritalStatus, m_pfNo, m_monthlySalary, m_homeAddress,
427 m_homePostcode, m_homeCity, m_homeState, m_positionGrade, m_position,
428 m_officeAddress, m_officePostcode, m_officeCity, m_officeState,
429 m_homeNumber, m_phoneNumber
430     FROM tb_member
431     WHERE m_memberNo = '$memberNo'";
432 $result = mysqli_query($con, $sql);
433
434 if ($row = mysqli_fetch_assoc($result)) {
435     $memberName = htmlspecialchars($row['m_name']);
436     $memberIC = htmlspecialchars($row['m_ic']);
437     $memberEmail = htmlspecialchars($row['m_email']);
438     $memberGender = $row['m_gender'];
439     $memberReligion = ($row['m_religion']);
440     $memberRace = ($row['m_race']);
441     $memberMaritalStatus = ($row['m_maritalStatus']);
442     $memberpfNo = htmlspecialchars($row['m_pfNo']);
443     $memberHomeAdd = htmlspecialchars($row['m_homeAddress']);
444     $memberHomePostcode = htmlspecialchars($row['m_homePostcode']);
445     $memberHomeCity = htmlspecialchars($row['m_homeCity']);
446     $memberHomeState = htmlspecialchars($row['m_homeState']);
447     $memberPositionGrade = htmlspecialchars($row['m_positionGrade']);
448     $memberPosition = htmlspecialchars($row['m_position']);
449     $memberOfficeAdd = htmlspecialchars($row['m_officeAddress']);
450     $memberOfficePostcode =
451     htmlspecialchars($row['m_officePostcode']);
452     $memberOfficeCity = htmlspecialchars($row['m_officeCity']);
453     $memberOfficeState = htmlspecialchars($row['m_officeState']);
454     $memberHomeTele = htmlspecialchars($row['m_homeNumber']);
455     $memberPhoneTele = htmlspecialchars($row['m_phoneNumber']);
456     $memberMonthlySalary = htmlspecialchars($row['m_monthlySalary']);
457
458 }
459 }
460
461 // Modify data in tb_member
462 if (!empty($memberNo)) {
463     $sql = "UPDATE tb_member
464             SET m_name = '$nama',
465                 m_email = '$email',
466                 m_gender = '$selectedGender',
467                 m_religion = '$selectedReligion',

```

```

468     m_race = '$selectedRace',
469     m_maritalStatus = '$selectedMaritalRace',
470     m_monthlySalary = '$selectedMonthlySalary',
471
472     m_homeAddress = '$selectedHomeAdd',
473     m_homePostcode = '$selectedHomePostcode',
474     m_homeCity = '$selectedHomeCity',
475     m_homeState = '$selectedHomeState',
476
477     m_position = '$selectedPosition',
478     m_positionGrade = '$selectedPositionGrade',
479
480     m_officeAddress = '$selectedOfficeAdd',
481     m_officePostcode = '$selectedOfficePostcode',
482     m_officeCity = '$selectedOfficeCity',
483     m_officeState = '$selectedOfficeState',
484
485     m_phoneNumber = '$selectedNoTeleBim',
486     m_homeNumber = '$selectedNoTelePej'
487     WHERE m_memberNo = '$memberNo'";
488
489 } else {
490     echo "Error: Missing member number.";
491 }
492
493 //Retrieve data from tb_guarantor for guarantor 1 details
494 if ($loanApplicationID !== null) {
495     $sql = "SELECT g_memberNo, g_signature, m.m_name, m.m_ic,
496 m.m_pfNo
497     FROM tb_guarantor g
498     LEFT JOIN tb_member m ON g.g_memberNo = m.m_memberNo
499     WHERE g.g_loanApplicationID = '$loanApplicationID'";
500     $result = mysqli_query($con, $sql);
501
502     if ($row = mysqli_fetch_assoc($result)) {
503         $memberNo1 = htmlspecialchars($row['g_memberNo']);
504         $signaturePenjamin1 = htmlspecialchars($row['g_signature']);
505         $namePenjamin1 = htmlspecialchars($row['m_name']);
506         $icPenjamin1 = htmlspecialchars($row['m_ic']);
507         $pfNoPenjamin1 = htmlspecialchars($row['m_pfNo']);
508     }
509 }
510
511 //Retrieve data from tb_guarantor for guarantor 2 details
512 if ($guarantorID2 !== null) {
513     $sql = "SELECT g_memberNo, g_signature, m.m_name, m.m_ic,
514 m.m_pfNo
515     FROM tb_guarantor g
516     LEFT JOIN tb_member m ON g.g_memberNo = m.m_memberNo
517     WHERE g.g_loanApplicationID = '$loanApplicationID' AND
518 g.g_guarantorID = '$guarantorID2'";
519     $result = mysqli_query($con, $sql);
520
521     if ($row = mysqli_fetch_assoc($result)) {
522         $memberNo2 = htmlspecialchars($row['g_memberNo']);
523         $signaturePenjamin2 = htmlspecialchars($row['g_signature']);
524         $namePenjamin2 = htmlspecialchars($row['m_name']);

```

```

525         $icPenjamin2 = htmlspecialchars($row['m_ic']);
526         $pfNoPenjamin2 = htmlspecialchars($row['m_pfNo']);
527     }
528 }
529
530 // Modify data in tb_guarantor for guarantor 1
531 if (!empty($memberNo1)) {
532     $update_stmt1 = mysqli_prepare
533     ($con, "UPDATE tb_guarantor
534         SET g_memberNo = ?, g_signature = ?
535         WHERE g_guarantorID = ?");
536     mysqli_stmt_bind_param($update_stmt1, "sss", $memberNo1,
537     $fileSignPenjamin1, $guarantorID1);
538     if (!mysqli_stmt_execute($update_stmt1)) {
539         throw new Exception("Failed to update guarantor 1: " .
540     mysqli_stmt_error($update_stmt1));
541     }
542     mysqli_stmt_close($update_stmt1);
543 }
544
545 // Modify data in tb_guarantor for guarantor 2
546 if (!empty($memberNo2)) {
547     $update_stmt2 = mysqli_prepare
548     ($con, "UPDATE tb_guarantor
549         SET g_memberNo = ?, g_signature = ?
550         WHERE g_guarantorID = ?");
551     mysqli_stmt_bind_param($update_stmt2, "sss", $memberNo2,
552     $fileSignPenjamin2, $guarantorID2);
553     if (!mysqli_stmt_execute($update_stmt2)) {
554         throw new Exception("Failed to update guarantor 2: " .
555     mysqli_stmt_error($update_stmt2));
556     }
557     mysqli_stmt_close($update_stmt2);
558 }
559
560
561
562
563
564
565

```

Use Case: View Financial Status

```

1 //Member Main Page
2 //Retrieve data from tb_financial for member share information
3 SELECT * FROM tb_financial
4             WHERE tb_financial.f_memberNo = '$u_id';
5
6 //Retrieve data from tb_loan and tb_ltype for member loan information
7 SELECT
8             tb_ltype.lt_desc AS loanName,
9             COALESCE(SUM(tb_loan.l_loanPayable)) AS totalLoan
10            FROM tb_ltype
11            LEFT JOIN tb_loan

```

```

12      ON tb_loan.l_loanType=tb_ltype.lt_lid
13      AND tb_loan.l_memberNo='$_u_id'
14          AND tb_loan.l_status=3
15      GROUP BY tb_ltype.lt_desc
16          ORDER BY tb_ltype.lt_desc;
17
18 //Retrieve data from tb_banner for advertisement
19 SELECT * FROM tb_banner;
20
21 //Salary Deduction Record Page
22 //Count number of desired records from tb_transaction for member
23 salary deduction record page pagination
24 SELECT COUNT(*) AS total FROM tb_transaction
25             WHERE t_memberNo='$_u_id' AND t_transactionType IN
26 (1,2,3,4,5);
27
28 //Retrieve data from tb_transaction, tb_ttype, and tb_rmonth for
29 member salary deduction record
30 SELECT tb_transaction.*,
31             tb_ttype.tt_desc AS type,
32             tb_rmonth.rm_desc AS month
33     FROM tb_transaction
34     LEFT JOIN tb_ttype ON
35 tb_transaction.t_transactionType=tb_ttype.tt_id
36     LEFT JOIN tb_rmonth ON tb_transaction.t_month=tb_rmonth.rm_id
37     WHERE tb_transaction.t_memberNo = '$_u_id' AND
38 tb_transaction.t_transactionType IN (1,2,3,4,5)
39         ORDER BY tb_transaction.t_transactionDate DESC
40         LIMIT $records_per_page OFFSET $offset;
41
42 //Loan Repayment Record Page
43 //Count number of desired records from tb_transaction for member loan
44 repayment record page pagination
45 SELECT COUNT(*) AS total FROM tb_transaction
46             WHERE t_memberNo='$_u_id' AND t_transactionType IN
47 (6,7,8,9,10,11,12);
48
49 //Retrieve data from tb_transaction, tb_ttype, and tb_rmonth for
50 member loan repayment transaction records
51 SELECT tb_transaction.*,
52             tb_ttype.tt_desc AS type,
53             tb_rmonth.rm_desc AS month
54     FROM tb_transaction
55     LEFT JOIN tb_ttype ON
56 tb_transaction.t_transactionType=tb_ttype.tt_id
57     LEFT JOIN tb_rmonth ON tb_transaction.t_month=tb_rmonth.rm_id
58     WHERE tb_transaction.t_memberNo = '$_u_id' AND
59 tb_transaction.t_transactionType IN (6,7,8,9,10,11,12)
60         ORDER BY tb_transaction.t_transactionDate DESC
61         LIMIT $records_per_page OFFSET $offset;
62

```

Use Case: Update Member Profile

1	//Member Profile Page
---	-----------------------

```

2 //Retrieve data from tb_member, tb_ugender, tb_urace, tb_ureligion,
3 tb_umaritalstatus, tb_homeState, and tb_officeState for member
4 personal information
5 SELECT tb_member.*,
6         tb_ugender.ug_desc AS gender,
7         tb_urace.ur_desc AS race,
8         tb_ureligion.ua_desc AS religion,
9         tb_umaritalstatus.um_desc AS maritalstatus,
10        tb_homeState.st_desc AS homeState,
11        tb_officeState.st_desc AS officeState,
12        tb_member.m_memberApplicationID
13    FROM tb_member
14    LEFT JOIN tb_ugender ON tb_member.m_gender=tb_ugender.ug_gid
15    LEFT JOIN tb_urace ON tb_member.m_race=tb_urace.ur_rid
16    LEFT JOIN tb_ureligion ON
17        tb_member.m_religion=tb_ureligion.ua_rid
18        LEFT JOIN tb_umaritalstatus ON
19        tb_member.m_maritalStatus=tb_umaritalstatus.um_mid
20        LEFT JOIN tb_homeState ON
21        tb_member.m_homeState=tb_homeState.st_id
22        LEFT JOIN tb_officeState ON
23        tb_member.m_officeState=tb_officeState.st_id
24        WHERE tb_member.m_memberNo = '$u_id';
25
26 //Retrieve data from tb_heir and tb_hrelation for member's heirs
27 information
28 SELECT tb_heir.* ,
29         tb_hrelation.hr_desc AS heirrelation
30     FROM tb_heir
31     LEFT JOIN tb_hrelation ON tb_heir.h_relationWithMember
32 = tb_hrelation.hr_rid
33         WHERE h_memberApplicationID = '$memberApplicationID'
34 ORDER BY h_heirID;
35
36 //Update Member Personal Information
37 //Retrieve data from tb_member, tb_ugender, tb_urace, tb_ureligion,
38 tb_umaritalstatus, tb_homeState, and tb_officeState for updating
39 member personal information
40 SELECT tb_member.*,
41         tb_ugender.ug_desc AS gender,
42         tb_urace.ur_desc AS race,
43         tb_ureligion.ua_desc AS religion,
44         tb_umaritalstatus.um_desc AS maritalstatus,
45         tb_homeState.st_desc AS homeState,
46         tb_officeState.st_desc AS officeState,
47         tb_member.m_memberApplicationID
48    FROM tb_member
49    LEFT JOIN tb_ugender ON tb_member.m_gender=tb_ugender.ug_gid
50    LEFT JOIN tb_urace ON tb_member.m_race=tb_urace.ur_rid
51    LEFT JOIN tb_ureligion ON
52        tb_member.m_religion=tb_ureligion.ua_rid
53        LEFT JOIN tb_umaritalstatus ON
54        tb_member.m_maritalStatus=tb_umaritalstatus.um_mid
55        LEFT JOIN tb_homeState ON
56        tb_member.m_homeState=tb_homeState.st_id
57        LEFT JOIN tb_officeState ON
58        tb_member.m_officeState=tb_officeState.st_id
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162

```

```

63      WHERE tb_member.m_memberNo = '$u_id';
64
65 //Update member personal information
66 UPDATE tb_member
67   SET
68     m_name = '$name',
69     m_religion = '$agama',
70     m_race = '$race',
71     m_maritalStatus = '$maritalstatus',
72     m_email = '$emel',
73     m_homeAddress = '$homeaddress',
74     m_homePostcode = '$homepostcode',
75     m_homeCity = '$homecity',
76     m_homeState = '$homestate',
77     m_position = '$position',
78     m_positionGrade = '$grade',
79     m_officeAddress = '$officeaddress',
80     m_officePostcode = '$officepostcode',
81     m_officeCity = '$officecity',
82     m_officeState = '$officestate',
83     m_faxNumber = '$faxphonenum',
84     m_phoneNumber = '$phonenum',
85     m_homeNumber = '$homephonenum',
86     m_monthlySalary = '$monthlysalary',
87     WHERE m_memberNo='$u_id';
88
89 //Update Member's Heirs Information
90 //Retrieve data from tb_member, tb_ugender, tb_urace, tb ureligion,
91 tb_umaritalstatus, tb_homeState, and tb_officeState for member
92 personal information
93 SELECT tb_member.*,
94           tb_ugender.ug_desc AS gender,
95           tb_urace.ur_desc AS race,
96           tb_ureligion.ua_desc AS religion,
97           tb_umaritalstatus.um_desc AS maritalstatus,
98           tb_homeState.st_desc AS homeState,
99           tb_officeState.st_desc AS officeState,
100          tb_member.m_memberApplicationID
101         FROM tb_member
102         LEFT JOIN tb_ugender ON
103          tb_member.m_gender=tb_ugender.ug_gid
104         LEFT JOIN tb_urace ON tb_member.m_race=tb_urace.ur_rid
105         LEFT JOIN tb_ureligion ON
106          tb_member.m_religion=tb_ureligion.ua_rid
107         LEFT JOIN tb_umaritalstatus ON
108          tb_member.m_maritalStatus=tb_umaritalstatus.um_mid
109         LEFT JOIN tb_homeState ON
110          tb_member.m_homeState=tb_homeState.st_id
111         LEFT JOIN tb_officeState ON
112          tb_member.m_officeState=tb_officeState.st_id
113          WHERE tb_member.m_memberNo = '$u_id';
114
115 //Retrieve data from tb_heir, and tb_hrelation for updating member's
116 heirs information
117 SELECT tb_heir.*, tb_hrelation.hr_desc AS heirrelation
118           FROM tb_heir
119

```

```

120      LEFT JOIN tb_hrelation ON tb_heir.h_relationWithMember =
121          tb_hrelation.hr_rid
122          WHERE h_memberApplicationID = '$memberApplicationID'
123          ORDER BY h_heirID;
124
125 //Update heir information
126 UPDATE tb_heir SET h_name = '$heirName', h_ic = '$heirIC',
127 h_relationWithMember = '$heirRelation'
128          WHERE h_heirID = '$heirID';
129
130 SELECT * FROM tb_heir WHERE h_memberApplicationID =
131 '$memberApplicationID' AND h_ic = '$heirIC'
132
133 //Add new heir
134 INSERT INTO tb_heir (h_memberApplicationID, h_name, h_ic,
135 h_relationWithMember)
136          VALUES ('$memberApplicationID', '$heirName',
137 '$heirIC', '$heirRelation');
138
139 //Delete existing heir
140 DELETE FROM tb_heir WHERE h_heirID = '$heirId';

```

Admin Dashboard

```

1 //Count number of desired records from tb_member and tb_loan for
2 admin dashboard
3 SELECT COUNT(*) AS total_members FROM tb_member WHERE m_status='3';
4
5 SELECT COUNT(*) AS pending_members FROM tb_member WHERE m_status='1';
6
7 SELECT COUNT(DISTINCT l_memberNo) AS loan_borrowers FROM tb_loan
8 WHERE l_status='3';
9
10 SELECT COUNT(*) AS pending_loans FROM tb_loan WHERE l_status='1';
11
12 SELECT COUNT(*) AS count FROM tb_member WHERE m_approvalDate BETWEEN
13 '$start_date' AND '$end_date' AND m_status='3';
14
15 SELECT COUNT(*) AS count FROM tb_loan WHERE l_approvalDate BETWEEN
16 '$start_date' AND '$end_date' AND l_status='3';
17
18 SELECT COUNT(*) AS pending_member_applications FROM tb_member WHERE
19 m_status='1' AND m_applicationDate BETWEEN '$currentYear-01-01' AND
20 '$currentYear-12-31';
21
22 SELECT COUNT(*) AS rejected_member_applications FROM tb_member WHERE
23 m_status='2' AND m_approvalDate BETWEEN '$currentYear-01-01' AND
24 '$currentYear-12-31';
25
26 SELECT COUNT(*) AS approved_member_applications FROM tb_member WHERE
27 m_status='3' AND m_approvalDate BETWEEN '$currentYear-01-01' AND
28 '$currentYear-12-31';
29
30

```

```

31 SELECT COUNT(*) AS pending_loan_applications FROM tb_loan WHERE
32 l_status='1' AND l_applicationDate BETWEEN '$currentYear-01-01' AND
33 '$currentYear-12-31';
34
35 SELECT COUNT(*) AS rejected_loan_applications FROM tb_loan WHERE
36 l_status='2' AND l_approvalDate BETWEEN '$currentYear-01-01' AND
37 '$currentYear-12-31';
38
39 SELECT COUNT(*) AS approved_loan_applications FROM tb_loan WHERE
40 l_status='3' AND l_approvalDate BETWEEN '$currentYear-01-01' AND
41 '$currentYear-12-31';
42
43 SELECT COUNT(*) AS al_bai_loan_applications FROM tb_loan WHERE
44 l_loanType='1' AND l_status='3' AND l_approvalDate BETWEEN
45 '$currentYear-01-01' AND '$currentYear-12-31';
46
47 SELECT COUNT(*) AS al_innah_loan_applications FROM tb_loan WHERE
48 l_loanType='2' AND l_status='3' AND l_approvalDate BETWEEN
49 '$currentYear-01-01' AND '$currentYear-12-31';
50
51 SELECT COUNT(*) AS kenderaan_loan_applications FROM tb_loan WHERE
52 l_loanType='3' AND l_status='3' AND l_approvalDate BETWEEN
53 '$currentYear-01-01' AND '$currentYear-12-31';
54
55 SELECT COUNT(*) AS road_tax_loan_applications FROM tb_loan WHERE
56 l_loanType='4' AND l_status='3' AND l_approvalDate BETWEEN
57 '$currentYear-01-01' AND '$currentYear-12-31';
58
59 SELECT COUNT(*) AS khas_loan_applications FROM tb_loan WHERE
60 l_loanType='5' AND l_status='3' AND l_approvalDate BETWEEN
61 '$currentYear-01-01' AND '$currentYear-12-31';
62
63 SELECT COUNT(*) AS karnival_loan_applications FROM tb_loan WHERE
64 l_loanType='6' AND l_status='3' AND l_approvalDate BETWEEN
65 '$currentYear-01-01' AND '$currentYear-12-31';
66
67 SELECT COUNT(*) AS al_qadrul_loan_applications FROM tb_loan WHERE
68 l_loanType='7' AND l_status='3' AND l_approvalDate BETWEEN
69 '$currentYear-01-01' AND '$currentYear-12-31';
70
71

```

Use Case: Approve Membership

```

1 // Select the member with the status ("Sedang Diproses")
2 "SELECT
3     tb_member.m_memberApplicationID,
4     tb_member.m_pfNo,
5     tb_member.m_name,
6     DATE_FORMAT(tb_member.m_applicationDate, '%d-%m-%Y') AS
7     formattedDate
8
9
10"

```

```

11    FROM tb_member
12    WHERE m_status = 1";
13
14
15 // Get total records count for pagination
16 "SELECT COUNT(*) FROM tb_member WHERE m_status = 1";
17
18
19 // Select the member and display the details
20 "SELECT * FROM tb_member
21
22     LEFT JOIN tb_status ON tb_member.m_status = tb_status.s_sid
23     LEFT JOIN tb_ugender ON tb_member.m_gender = tb_ugender.ug_gid
24     LEFT JOIN tb_ureligion ON tb_member.m_religion =
25 tb_ureligion.ua_rid
26
27     LEFT JOIN tb_urace ON tb_member.m_race = tb_urace.ur_rid
28     LEFT JOIN tb_umaritalstatus ON tb_member.m_maritalStatus =
29 tb_umaritalstatus.um_mid
30
31     WHERE tb_member.m_memberApplicationID = '$mApplicationID'";
32
33
34 // Select SQL query with DATE_FORMAT for m_approvalDate to
35 // display in a list
36
37 $sql = "SELECT m_memberNo, m_pfNo, m_name,
38
39     DATE_FORMAT(m_approvalDate, '%d-%m-%Y') AS formattedDate
40
41     FROM tb_member
42
43     WHERE m_status = 3";
44
45
46 // Display the member details
47 "SELECT * FROM tb_member
48
49     LEFT JOIN tb_ugender ON tb_member.m_gender = tb_ugender.ug_gid
50     LEFT JOIN tb_ureligion ON tb_member.m_religion =
51 tb_ureligion.ua_rid
52
53     LEFT JOIN tb_urace ON tb_member.m_race = tb_urace.ur_rid
54     LEFT JOIN tb_umaritalstatus ON tb_member.m_maritalStatus =
55 tb_umaritalstatus.um_mid
56
57
58     WHERE m_memberNo = '$member_id'";
59

```

Use Case: Approve Loan Application

1	// Display the member details in loan list
2	
3	

```

4  SELECT tb_loan.l_loanApplicationID, tb_loan.l_memberNo,
5    tb_member.m_name,
6      DATE_FORMAT(tb_loan.l_applicationDate, '%d-%m-%Y') AS
7  formattedDate
8    FROM tb_loan
9      LEFT JOIN tb_member ON tb_loan.l_memberNo =
10     tb_member.m_memberNo
11       WHERE l_status = 1;
12
13
14
15
16 // Display the loan applicant details
17 "SELECT l.* , m.m_name, m.m_pfNo, lt.lt_desc, b.lb_id, b.lb_desc
18   FROM tb_loan l
19     LEFT JOIN tb_member m ON l.l_memberNo = m.m_memberNo
20       LEFT JOIN tb_ltype lt ON l.l_loanType = lt.lt_lid
21         LEFT JOIN tb_lbank b ON l.l_bankName = b.lb_id
22           WHERE l.l_loanApplicationID = '$lApplicationID'";
23
24
25 // Get the information of guarantor for loan application
26 "SELECT g.g_memberNo, g.g_signature FROM tb_guarantor g WHERE
27 g.g_loanApplicationID = '$lApplicationID' LIMIT 2"
28
29
30
31 // Fetch member details for guarantor 1
32 "SELECT m.m_name, m.m_ic, m.m_pfNo, m.m_memberNo FROM tb_member m
33 WHERE m.m_memberNo = '$guarantorID1'";
34
35 // Fetch member details for guarantor 2
36 "SELECT m.m_name, m.m_ic, m.m_pfNo, m.m_memberNo FROM tb_member m
37 WHERE m.m_memberNo = '$guarantorID2'";
38
39
40 // Display the loan applicant list
41 "SELECT * FROM tb_loan
42   LEFT JOIN tb_member ON tb_loan.l_memberNo =
43     tb_member.m_memberNo
44       WHERE l_status = 3";
45
46
47
48
49 // Display the loan applicant's details
50 "SELECT tb_loan.*, tb_member.m_name, tb_member.m_pfNo,
51 tb_ltype.lt_desc, tb_lbank.lb_desc
52   FROM tb_loan
53     LEFT JOIN tb_member ON tb_loan.l_memberNo =
54       tb_member.m_memberNo
55         LEFT JOIN tb_ltype ON tb_loan.l_loanType = tb_ltype.lt_lid
56           LEFT JOIN tb_lbank ON tb_loan.l_bankName = tb_lbank.lb_id
57             WHERE tb_loan.l_loanApplicationID = '$loan_id'";
58
59
60

```

```

61
62 // Get the information of guarantor for loan application
63
64 "SELECT g.g_memberNo, g.g_signature FROM tb_guarantor g WHERE
65 g.g_loanApplicationID = '$loan_id' LIMIT 2";
66
67
68 // Fetch member details for guarantor 1
69 "SELECT m.m_name, m.m_ic, m.m_pfNo, m.m_memberNo FROM tb_member m
70 WHERE m.m_memberNo = '$guarantorID1'";
71
72
73 // Fetch member details for guarantor 2
74 "SELECT m.m_name, m.m_ic, m.m_pfNo, m.m_memberNo FROM tb_member m
75 WHERE m.m_memberNo = '$guarantorID2'";
76
77
78
79
80
81

```

Use Case: Manage System

```

1 -----
2 --- Transaction
3 -----
4 --- Salary Deduction
5 --- Retrieve financial list of all members
6 SELECT tb_financial.*, tb_member.m_name
7     FROM tb_financial
8     INNER JOIN tb_member
9         ON tb_financial.f_memberNo=tb_member.m_memberNo;
10
11 --- Retrieve the sum of active loan by type of each member
12 SELECT l_memberNo, l_loanType, COALESCE(SUM(l_loanPayable), 0) AS
13 totalLoan
14     FROM tb_loan
15     WHERE l_status = 3
16     GROUP BY l_memberNo, l_loanType
17     ORDER BY l_memberNo, l_loanType;
18
19 --- Retrieve the month for selection
20 SELECT * FROM tb_rmonth;
21
22 --- Retrieve the loan types for display
23 SELECT * FROM tb_ltype
24     ORDER BY lt_lid;
25
26 --- Retrieve policy information for calculation
27 SELECT p_minShareCapital, p_salaryDeductionForSaving,
28 p_salaryDeductionForMemberFund
29     FROM tb_policies
30     ORDER BY p_policyID DESC
31     LIMIT 1;

```

```

32
33 -- Retrieve the financial information by member
34 SELECT * FROM tb_financial WHERE f_memberNo = $memberNo;
35
36 -- Retrieve member information by member
37 SELECT * FROM tb_member WHERE m_memberNo = $memberNo;
38
39 -- Retrieve active loan information by member
40 SELECT tb_loan.*, tb_ltype.lt_desc
41   FROM tb_loan
42   JOIN tb_ltype ON tb_loan.l_loanType = tb_ltype.lt_lid
43 WHERE tb_loan.l_memberNo = $memberNo AND tb_loan.l_status = 3;
44
45 -- Check if the salary is deducted for the selected month
46 SELECT COUNT(t_transactionID) FROM tb_transaction
47   WHERE t_memberNo = $memberNo
48   AND t_month = $f_month
49   AND t_year = $f_year
50   AND t_method = 'Potongan Gaji';
51
52 -- Backend process after submission
53 -- Update the financial status after calculation
54 UPDATE tb_financial
55   SET f_shareCapital = f_shareCapital + '$shareCapitalChange',
56       f_feeCapital = f_feeCapital + '$feeCapitalChange',
57       f_fixedSaving = f_fixedSaving + '$fixedSavingChange',
58       f_memberSaving = f_memberSaving + '$memberSavingChange',
59       f_memberFund = f_memberFund + '$memberFundChange'
60   WHERE f_memberNo = '$memberNo';
61
62 -- Record the transaction to the transaction table
63 INSERT INTO tb_transaction(t_transactionType, t_method,
64 t_transactionAmt, t_month, t_year, t_desc, t_memberNo, t_adminID)
65   VALUES ('1', 'Potongan Gaji', '$difference', '$f_month', '$f_year',
66 'Potongan Gaji', '$memberNo', '$admin_id');
67
68 INSERT INTO tb_transaction(t_transactionType, t_method,
69 t_transactionAmt, t_month, t_year, t_desc, t_memberNo, t_adminID)
70   VALUES ('3', 'Potongan Gaji', '$difference', '$f_month', '$f_year',
71 'Potongan Gaji', '$memberNo', '$admin_id');
72
73 INSERT INTO tb_transaction(t_transactionType, t_method,
74 t_transactionAmt, t_month, t_year, t_desc, t_memberNo, t_adminID)
75   VALUES ('4', 'Potongan Gaji', '$difference', '$f_month', '$f_year',
76 'Potongan Gaji', '$memberNo', '$admin_id');
77
78 INSERT INTO tb_transaction(t_transactionType, t_method,
79 t_transactionAmt, t_month, t_year, t_desc, t_memberNo, t_adminID)
80   VALUES ('$transactionType', 'Potongan Gaji', $difference,
81 '$f_month', '$f_year', '$desc', '$memberNo', '$admin_id');
82
83 -- Update loan table
84 UPDATE tb_loan
85   SET l_loanPayable = $newLoanPayable,
86       l_status = $status
87   WHERE l_loanApplicationID = $loanID;
88

```

```

89  -- Other transaction
90  -- Retrieve the financial and member information by member
91  SELECT tb_financial.*, tb_member.m_name, tb_member.m_ic,
92  tb_member.m_pfNo
93  FROM tb_financial
94  INNER JOIN tb_member
95  ON tb_financial.f_memberNo=tb_member.m_memberNo
96  WHERE f_memberNo = '$memberNo';
97
98  -- Retrieve active loan information by member
99  SELECT tb_loan.*, tb_ltype.lt_desc
100  FROM tb_loan
101  JOIN tb_ltype ON tb_loan.l_loanType = tb_ltype.lt_lid
102  WHERE tb_loan.l_memberNo = $memberNo AND tb_loan.l_status = 3;
103
104  -- Backend process after submission
105  -- Record the transaction to the transaction table
106  INSERT INTO tb_transaction (t_transactionType, t_method,
107  t_transactionAmt, t_month, t_year, t_desc, t_memberNo, t_adminID)
108  VALUES ('$type', 'Transaksi Tambahan', '$changeAmount',
109  '$currentMonth', '$currentYear', '$desc', '$memberNo', '$admin_id');
110
111  -- Update loan table
112  UPDATE tb_loan
113  SET l_loanPayable = '$newLoanPayable',
114  l_status = '$status'
115  WHERE l_loanApplicationID = '$loanID';
116
117  -- Retrieve loan types to calculate transaction type
118  SELECT l_loanType from tb_loan
119  WHERE l_loanApplicationID = '$loanID';
120
121  -- Record the transaction to the transaction table
122  INSERT INTO tb_transaction (t_transactionType, t_method,
123  t_transactionAmt, t_month, t_year, t_desc, t_memberNo, t_adminID)
124  VALUES ('$ttype', 'Transaksi Tambahan', '$paymentAmount',
125  '$currentMonth', '$currentYear', '$desc', '$memberNo', '$admin_id');
126
127  -- Update the financial table
128  UPDATE tb_financial
129  SET f_shareCapital = f_shareCapital + '$shareCapitalChange',
130  f_feeCapital = f_feeCapital + '$feeCapitalChange',
131  f_fixedSaving = f_fixedSaving + '$fixedSavingChange',
132  f_memberSaving = f_memberSaving + '$memberSavingChange',
133  f_memberFund = f_memberFund + '$memberFundChange'
134  WHERE f_memberNo = '$memberNo';
135
136  -----
137  -- Update Policies
138  -----
139  -- Retrieve the latest policy with newest ID
140  SELECT * FROM tb_policies
141  ORDER BY p_policyID DESC
142  LIMIT 1;
143
144  -- Backend process after submission
145  -- Insert new record

```

```

146 INSERT INTO tb_policies (
147     p_memberRegFee, p_minShareCapital, p_minFeeCapital,
148     p_minFixedSaving, p_minMemberFund, p_minMemberSaving, p_minOtherFees,
149     p_minShareCapitalForLoan, p_profitRate, p_maxInstallmentPeriod,
150     p_maxFinancingAmt,
151     p_salaryDeductionForSaving, p_salaryDeductionForMemberFund,
152     p_adminID)
153 VALUES (
154     '$f_memberRegFee', '$f_minShareCapital', '$f_minFeeCapital',
155     '$f_minFixedSaving', '$f_minMemberFund', '$f_minMemberSaving',
156     '$f_minOtherFees',
157     '$d_minShareCapitalForLoan', '$d_profitRate',
158     '$d_maxInstallmentPeriod', '$d_maxFinancingAmt',
159     '$d_salaryDeductionForSaving', '$d_salaryDeductionorMemberFund',
160     '$admin_id');
161
162 INSERT INTO tb_policies (
163     p_memberRegFee, p_minShareCapital, p_minFeeCapital,
164     p_minFixedSaving, p_minMemberFund, p_minMemberSaving, p_minOtherFees,
165     p_minShareCapitalForLoan, p_profitRate, p_maxInstallmentPeriod,
166     p_maxFinancingAmt,
167     p_salaryDeductionForSaving, p_salaryDeductionForMemberFund,
168     p_adminID)
169 VALUES (
170     '$d_memberRegFee', '$d_minShareCapital', '$d_minFeeCapital',
171     '$d_minFixedSaving', '$d_minMemberFund', '$d_minMemberSaving',
172     '$d_minOtherFees',
173     '$f_minShareCapitalForLoan', '$f_profitRate',
174     '$f_maxInstallmentPeriod', '$f_maxFinancingAmt',
175     '$d_salaryDeductionForSaving', '$d_salaryDeductionForMemberFund',
176     '$admin_id');
177
178 INSERT INTO tb_policies (
179     p_memberRegFee, p_minShareCapital, p_minFeeCapital,
180     p_minFixedSaving, p_minMemberFund, p_minMemberSaving, p_minOtherFees,
181     p_minShareCapitalForLoan, p_profitRate, p_maxInstallmentPeriod,
182     p_maxFinancingAmt,
183     p_salaryDeductionForSaving, p_salaryDeductionForMemberFund,
184     p_adminID)
185 VALUES (
186     '$d_memberRegFee', '$d_minShareCapital', '$d_minFeeCapital',
187     '$d_minFixedSaving', '$d_minMemberFund', '$d_minMemberSaving',
188     '$d_minOtherFees',
189     '$d_minShareCapitalForLoan', '$d_profitRate',
190     '$d_maxInstallmentPeriod', '$d_maxFinancingAmt',
191     '$f_salaryDeductionForSaving', '$f_salaryDeductionForMemberFund',
192     '$admin_id');
193
194 -----
195 -- Update Banner
196 -----
197 -- Retrieve banners that are active
198 SELECT * FROM tb_banner WHERE b_status = 1;
199
200 -- Backend process after upload
201 INSERT INTO tb_banner (b_banner, b_name, b_status, b_adminID)
202 VALUES ('$bannerPath', '$bannerName', '$b_status', '$admin_id');

```

```

203
204 -- Update banner to be displayed
205 -- Retrieve banners regardless of status with pagination
206 SELECT * FROM tb_banner
207     ORDER BY b_bannerID ASC
208     LIMIT $start_from, $records_per_page;
209
210 -- Count the number of banner for pagination calculation
211 SELECT COUNT(*) FROM tb_banner;
212
213 -- Backend processes
214 -- Retrieve banner to check existence before deleting
215 SELECT b_banner FROM tb_banner
216     WHERE b_bannerID = $bannerID;
217
218 -- Delete banner
219 DELETE FROM tb_banner
220     WHERE b_bannerID = $bannerID;
221
222 -- Update banner status
223 UPDATE tb_banner
224     SET b_status = $status
225     WHERE b_bannerID = $bannerID;
226
227 -- Check the number of active banners
228 SELECT COUNT(*) FROM tb_banner
229     WHERE b_status = 1;
230
231
232
233

```

Use Case: Generate Report

```

1 // Retrieve data from tb_rmonth for display month
2 if ($bulan && $tahun) {
3     $month_query = mysqli_query
4         ($con, "SELECT rm_desc
5             FROM tb_rmonth
6             WHERE rm_id = '$bulan'");
7     if ($month_query && mysqli_num_rows($month_query) > 0) {
8         $month_name = mysqli_fetch_assoc($month_query) ['rm_desc'];
9         $period = $month_name . " " . $tahun;
10    }
11 } elseif ($bulan) {
12     $month_query = mysqli_query
13         ($con, "SELECT rm_desc
14             FROM tb_rmonth
15             WHERE rm_id = '$bulan'");
16     if ($month_query && mysqli_num_rows($month_query) > 0) {
17         $month_name = mysqli_fetch_assoc($month_query) ['rm_desc'];
18         $period = $month_name;
19    }
20 } elseif ($tahun) {
21     $period = "Tahun " . $tahun;

```

```

22 }
23
24 // Retrieve data from tb_member
25 $member_sql = "SELECT COUNT(*) AS member_count
26         FROM tb_member $where_clause";
27 $member_result = mysqli_query($con, $member_sql);
28 if ($member_result) {
29     $member_data = mysqli_fetch_assoc($member_result);
30     $member_count = $member_data['member_count'];
31 }
32
33 // Retrieve data from tb_member group by status
34 $status_sql = "SELECT m_status, COUNT(*) AS status_count
35         FROM tb_member
36         $where_clause
37         GROUP BY m_status";
38 $status_result = mysqli_query($con, $status_sql);
39
40 if ($status_result) {
41     while ($status_row = mysqli_fetch_assoc($status_result)) {
42         switch ($status_row['m_status']) {
43             case 1: $member_pending = $status_row['status_count'];
44             break;
45             case 2: $member_rejected = $status_row['status_count'];
46             break;
47             case 3: $member_approved = $status_row['status_count'];
48             break;
49         }
50     }
51 }
52
53 // Retrieve data from tb_loan
54 $loan_where = str_replace('m_applicationDate', 'l_applicationDate',
55 $where_clause);
56 $loan_sql = "SELECT COUNT(*) AS loan_count
57         FROM tb_loan $loan_where";
58 $loan_result = mysqli_query($con, $loan_sql);
59 if ($loan_result) {
60     $loan_data = mysqli_fetch_assoc($loan_result);
61     $loan_count = $loan_data['loan_count'];
62 }
63
64 // Retrieve data from tb_loan group by status
65 $loan_status_sql = "SELECT l_status, COUNT(*) AS loan_status_count,
66         SUM(CASE WHEN l_status = 3 THEN l_appliedLoan
67 ELSE 0 END) AS total_approved_loans
68         FROM tb_loan $loan_where
69         GROUP BY l_status";
70 $loan_status_result = mysqli_query($con, $loan_status_sql);
71
72 if ($loan_status_result) {
73     while ($loan_status_row =
74 mysqli_fetch_assoc($loan_status_result)) {
75         switch ($loan_status_row['l_status']) {
76             case 1: $loan_pending =
77 $loan_status_row['loan_status_count']; break;
78

```

```

79         case 2: $loan_rejected =
80             $loan_status_row['loan_status_count']; break;
81         case 3: $loan_approved =
82             $loan_status_row['loan_status_count'];
83             $total_approved_loans =
84             $loan_status_row['total_approved_loans'];
85             break;
86     }
87 }
88
89 // Retrieve data from tb_loan group by type
90 $loan_type_sql = "SELECT l_loanType, COUNT(*) AS loan_type_count
91     FROM tb_loan
92     $loan_where
93     GROUP BY l_loanType";
94 $loan_type_result = mysqli_query($con, $loan_type_sql);
95
96 if ($loan_type_result) {
97     while ($loan_type_row = mysqli_fetch_assoc($loan_type_result)) {
98         switch ($loan_type_row['l_loanType']) {
99             case 1: $alBai = $loan_type_row['loan_type_count'];
100             break;
101             case 2: $alInnah = $loan_type_row['loan_type_count'];
102             break;
103             case 3: $baikPulihKenderaan =
104                 $loan_type_row['loan_type_count']; break;
105             case 4: $roadTax = $loan_type_row['loan_type_count'];
106             break;
107             case 5: $khas = $loan_type_row['loan_type_count']; break;
108             case 6: $karnival = $loan_type_row['loan_type_count'];
109             break;
110             case 7: $alQadrul = $loan_type_row['loan_type_count'];
111             break;
112         }
113     }
114 }
115
116
117 // Retrieve data from tb_transaction
118 $transaction_where = str_replace('m_applicationDate',
119 't_transactionDate', $where_clause);
120 $transactions_sql = "SELECT COUNT(*) AS transaction_count,
121     SUM(t_transactionAmt) AS transaction_total
122     FROM tb_transaction $transaction_where";
123 $transactions_result = mysqli_query($con, $transactions_sql);
124
125 if ($transactions_result && mysqli_num_rows($transactions_result) >
126 0) {
127     $transaction_data = mysqli_fetch_assoc($transactions_result);
128     $transaction_count = $transaction_data['transaction_count'] ?? 0;
129     $transaction_total = $transaction_data['transaction_total'] ?? 0;
130 } else {
131     $transaction_total = 0;
132 }
133
134 // Retrieve data from tb_policies
135

```

```

136 $policies_where = str_replace('m_applicationDate', 'p_dateUpdated',
137 $where_clause);
138 $policies_sql = "SELECT * FROM tb_policies $policies_where";
139 $policies_result = mysqli_query($con, $policies_sql);
140
141
142 // Retrieve data from tb_member
143 $member_sql = "SELECT
144     COUNT(*) as total_count,
145     SUM(CASE WHEN m_status = 1 THEN 1 ELSE 0 END) as
146 pending_count,
147     SUM(CASE WHEN m_status = 2 THEN 1 ELSE 0 END) as
148 rejected_count,
149     SUM(CASE WHEN m_status = 3 THEN 1 ELSE 0 END) as
150 approved_count
151     FROM tb_member" . createWhereClause($bulan, $tahun,
152 'm_applicationDate');
153
154     error_log("Member SQL: " . $member_sql);
155
156 $member_result = mysqli_query($con, $member_sql);
157 $member_data = mysqli_fetch_assoc($member_result);
158
159 $header = array('Status', 'Jumlah');
160 $data = array(
161     array('Diluluskan', $member_data['approved_count'] ?? 0),
162     array('Sedang Diproses', $member_data['pending_count'] ?? 0),
163     array('Ditolak', $member_data['rejected_count'] ?? 0),
164     array('JUMLAH PERMOHONAN', $member_data['total_count'] ?? 0)
165 );
166
167
168 // Retrieve data from tb_loan
169 $loan_sql = "SELECT
170     COUNT(*) as total_count,
171     SUM(CASE WHEN l_status = 1 THEN 1 ELSE 0 END) as
172 pending_count,
173     SUM(CASE WHEN l_status = 2 THEN 1 ELSE 0 END) as
174 rejected_count,
175     SUM(CASE WHEN l_status = 3 THEN 1 ELSE 0 END) as
176 approved_count,
177     SUM(l_appliedLoan) as total_amount
178     FROM tb_loan" . createWhereClause($bulan, $tahun,
179 'l_applicationDate');
180
181 $loan_result = mysqli_query($con, $loan_sql);
182 $loan_data = mysqli_fetch_assoc($loan_result);
183
184 $header = array('Status', 'Jumlah');
185 $data = array(
186     array('Diluluskan', $loan_data['approved_count'] ?? 0),
187     array('Sedang Diproses', $loan_data['pending_count'] ?? 0),
188     array('Ditolak', $loan_data['rejected_count'] ?? 0),
189     array('JUMLAH PERMOHONAN', $loan_data['total_count'] ?? 0)
190 );
191
192 $loan_type_sql = "SELECT

```

```

193     SUM(CASE WHEN l_loanType = 1 THEN 1 ELSE 0 END) as alBai,
194     SUM(CASE WHEN l_loanType = 2 THEN 1 ELSE 0 END) as alInnah,
195     SUM(CASE WHEN l_loanType = 3 THEN 1 ELSE 0 END) as
196 baikPulihKenderaan,
197     SUM(CASE WHEN l_loanType = 4 THEN 1 ELSE 0 END) as roadTax,
198     SUM(CASE WHEN l_loanType = 5 THEN 1 ELSE 0 END) as khas,
199     SUM(CASE WHEN l_loanType = 6 THEN 1 ELSE 0 END) as karnival,
200     SUM(CASE WHEN l_loanType = 7 THEN 1 ELSE 0 END) as alQadrul
201     FROM tb_loan" . createWhereClause($bulan, $tahun,
202 'l_applicationDate');

203
204     $loan_type_result = mysqli_query($con, $loan_type_sql);
205     $loan_type_data = mysqli_fetch_assoc($loan_type_result);
206
207     $header = array('Jenis Pinjaman', 'Jumlah');
208     $data = array(
209         array('Al-Bai', $loan_type_data['alBai'] ?? 0),
210         array('Al-Innah', $loan_type_data['alInnah'] ?? 0),
211         array('Baik Pulih Kenderaan',
212 $loan_type_data['baikPulihKenderaan'] ?? 0),
213             array('Road Tax dan Insurans', $loan_type_data['roadTax'] ?? 0),
214         array('Khas', $loan_type_data['khas'] ?? 0),
215         array('Karnival Musim Istimewa', $loan_type_data['karnival'] ?? 0),
216         array('Al-Qadrul Hassan', $loan_type_data['alQadrul'] ?? 0),
217         array('JUMLAH', array_sum($loan_type_data))
218     );
219
220
221
222 // Retrieve data from tb_transaction
223 $transaction_sql = "SELECT
224     COUNT(*) as transaction_count,
225     COALESCE(SUM(t_transactionAmt), 0) as transaction_total
226     FROM tb_transaction" . createWhereClause($bulan, $tahun,
227 't_transactionDate');

228
229     $transaction_result = mysqli_query($con, $transaction_sql);
230     $transaction_data = mysqli_fetch_assoc($transaction_result);
231
232     $header = array('Perkara', 'Jumlah');
233     $data = array(array('Bilangan Transaksi',
234 $transaction_data['transaction_count'] ?? 0),
235             array('JUMLAH AMAUN', 'RM ' .
236 number_format($transaction_data['transaction_total'] ?? 0, 2))
237     );
238
239
240 // Retrieve data from tb_policies
241 $policies_sql = "SELECT * FROM tb_policies";
242     if ($bulan && $tahun) {
243         $policies_sql .= " WHERE MONTH(p_dateUpdated) = '$bulan' AND
244 YEAR(p_dateUpdated) = '$tahun'";
245     }
246     $policies_sql .= " ORDER BY p_dateUpdated DESC LIMIT 1";
247
248     $policies_result = mysqli_query($con, $policies_sql);
249     $policy_data = mysqli_fetch_assoc($policies_result);

```

```

249
250     $header = array('Perkara', 'Nilai');
251     $data = array(
252         array('Yuran Pendaftaran Anggota', 'RM ' .
253             number_format($policy_data['p_memberRegFee'] ?? 0, 2)),
254             array('Modal Syer Minimum', 'RM ' .
255                 number_format($policy_data['p_minShareCapital'] ?? 0, 2)),
256                 array('Yuran Modal Minimum', 'RM ' .
257                     number_format($policy_data['p_minFeeCapital'] ?? 0, 2)),
258                     array('Yuran Tetap Minimum', 'RM ' .
259                         number_format($policy_data['p_minFixedSaving'] ?? 0, 2)),
260                             array('Simpanan Tetap Minimum', 'RM ' .
261                                 number_format($policy_data['p_minMemberFund'] ?? 0, 2)),
262                                     array('Tabung Kebajikan Minimum', 'RM ' .
263                                         number_format($policy_data['p_minMemberSaving'] ?? 0, 2)),
264                                         array('Jumlah Pembiayaan Maksimum', 'RM ' .
265                                             number_format($policy_data['p_maxFinancingAmt'] ?? 0, 2)),
266                                                 array('Kadar Keuntungan', ($policy_data['p_profitRate'] ?? 0) .
267 . '%')
268 );
269
270 // Insert data into tb_reportretrievallog
271 $sql = "INSERT INTO tb_reportretrievallog (r_retrievalDate, r_month,
272 r_year, r_adminID)
273     VALUES (CURRENT_TIMESTAMP(), ?, ?, ?)";
274
275 $stmt = $con->prepare($sql);
276 if (!$stmt) {
277     throw new Exception("Prepare failed: " . $con->error);
278 }
279
280 $stmt->bind_param("sss", $month, $year, $adminNo);
281 if (!$stmt->execute()) {
282     throw new Exception("Execute failed: " . $stmt->error);
283 }
284
285 if ($stmt->affected_rows > 0) {
286     echo json_encode(['status' => 'success']);
287 } else {
288     throw new Exception("No rows inserted");
289 }
290
291
292
293
294
295
296
297
298

```

4 Summary

The report details the conceptual and logical database design, interface design and SQL implementation of the KKK Online System that is proposed for Koperasi Kakitangan KADA Berhad to digitalise the existing manual system.

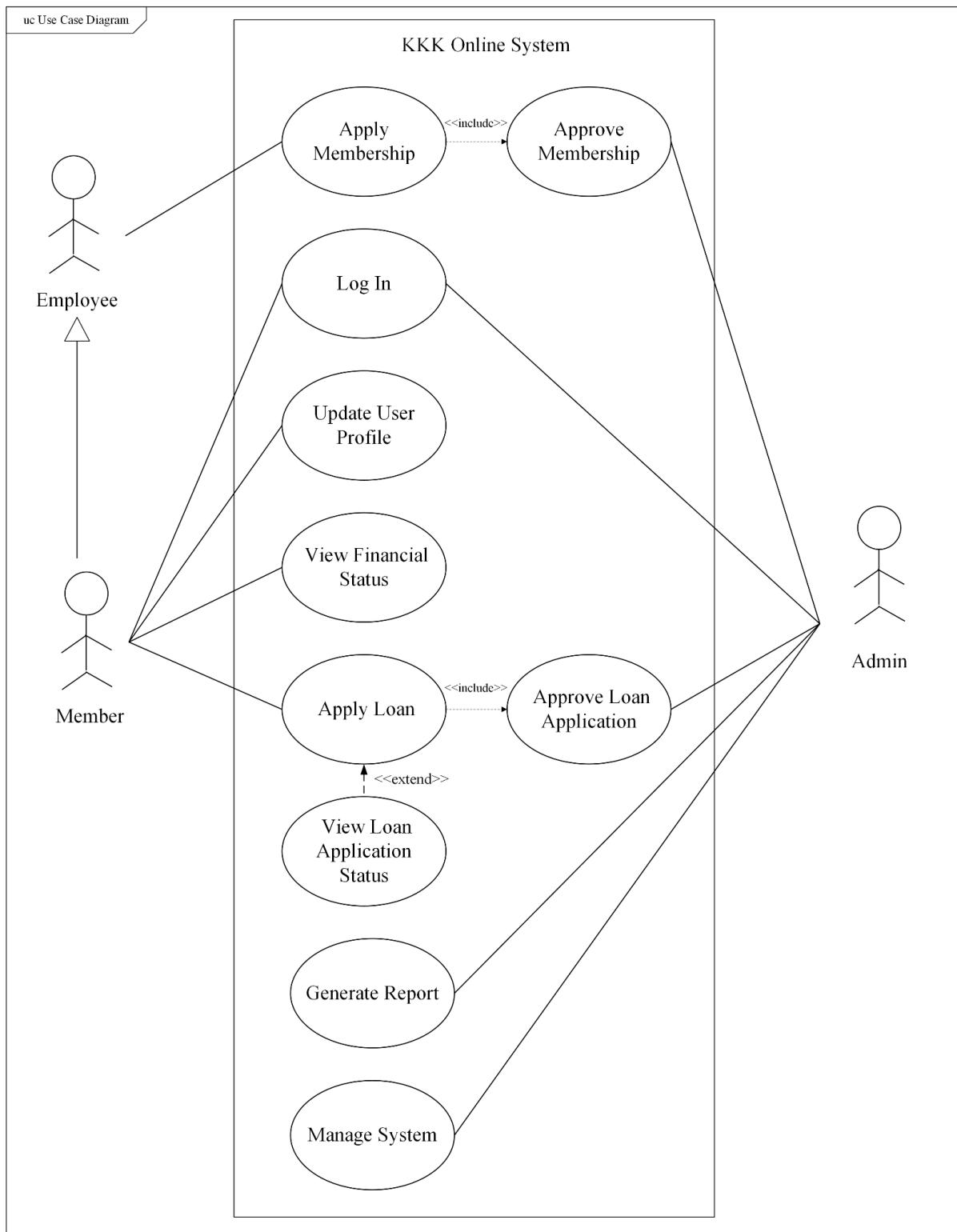
For the conceptual database design, an ERD was drawn, identifying the major entities according to the business requirements. The major entities identified included the admin, member, heir (of the member), loan, guarantor (of the loan applicant), financial status (of the member), transaction, policies, banner and report retrieval log.

In logical database design, a more detailed ERD was developed. To ensure data consistency, more reference tables are created such as user type, user gender, user religion, user race, user marital status, state, application status, loan type, bank and month. From the transformed ERD, relational schemas are derived and normalised.

After that, the SQL statements were written to create the database and tables, and queries were implemented in the actual system.

5 Appendices

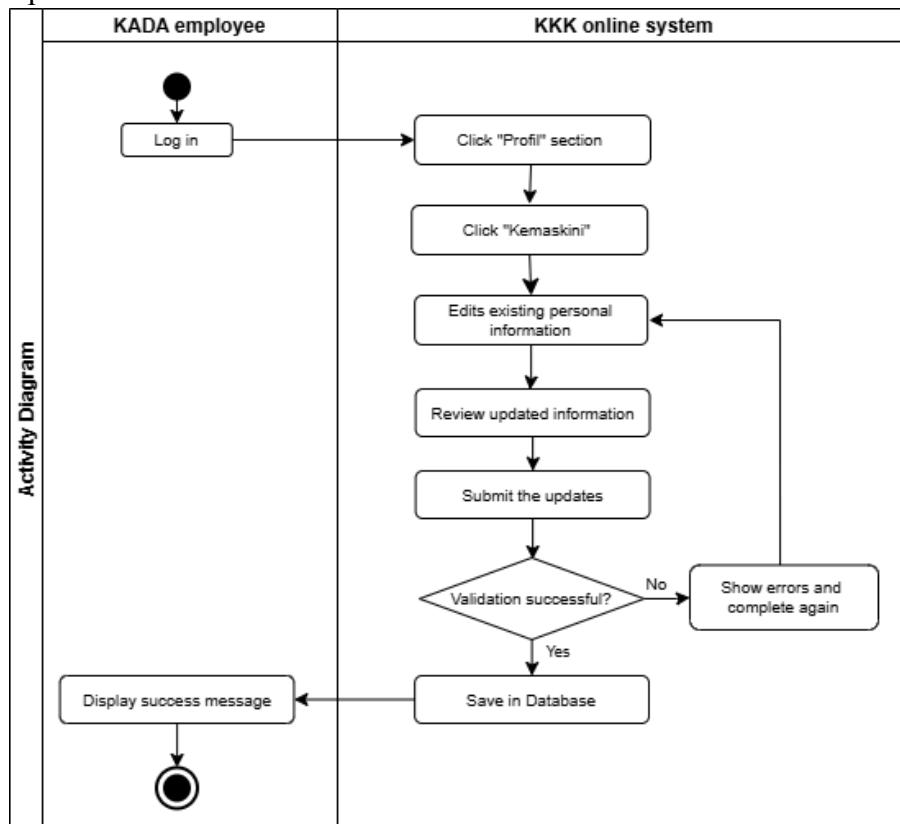
5.1 Use Case Diagram



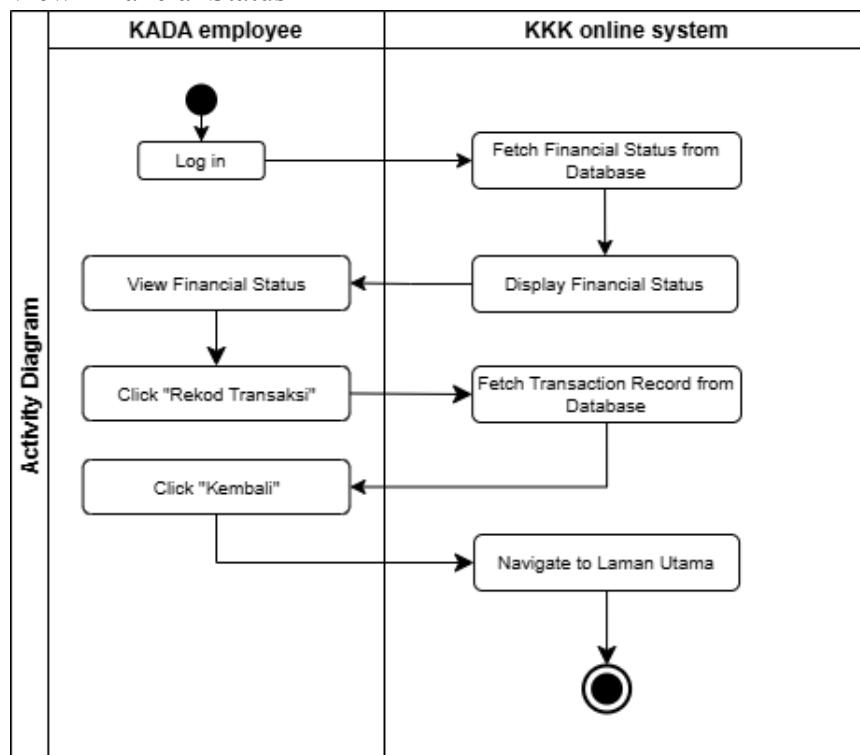
5.2 Activity Diagram

Member

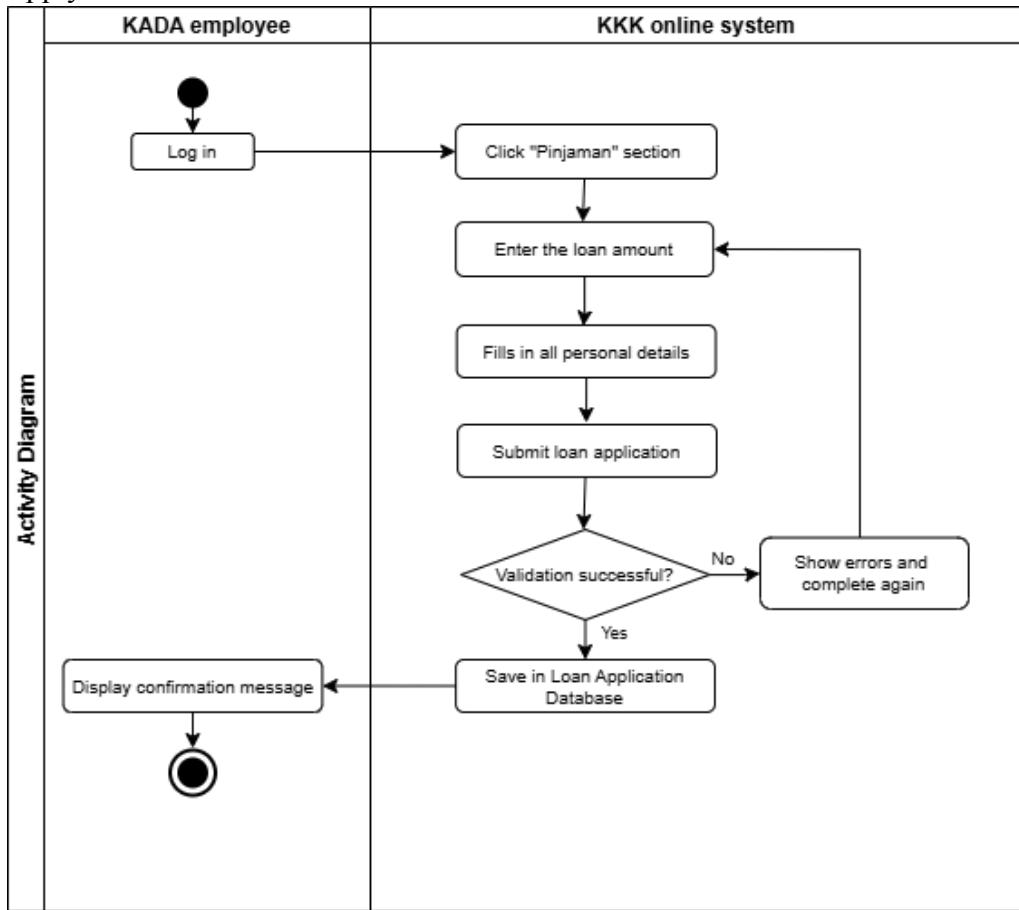
Update User Profile



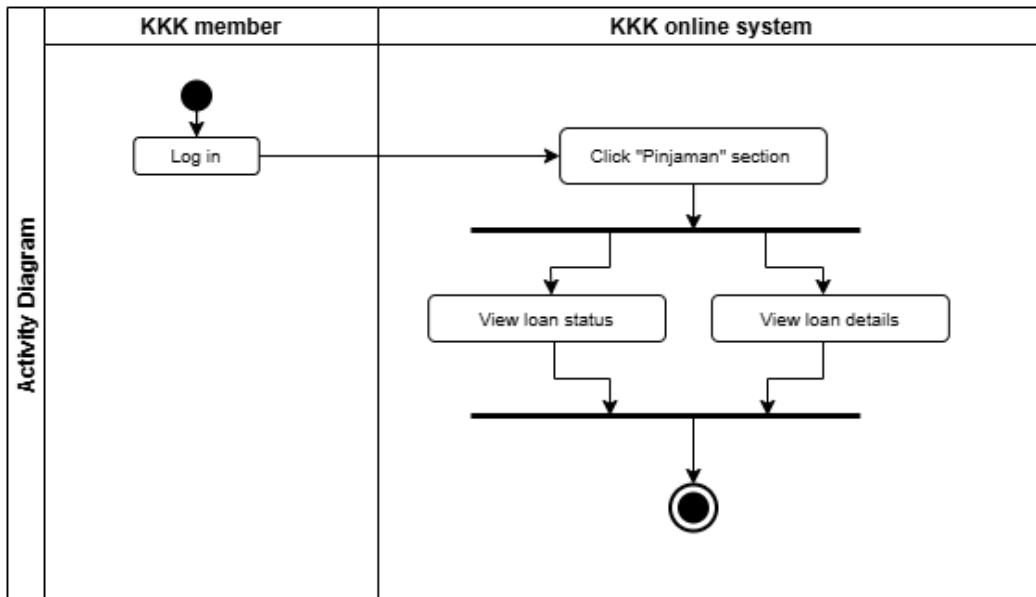
View Financial Status



Apply Loan

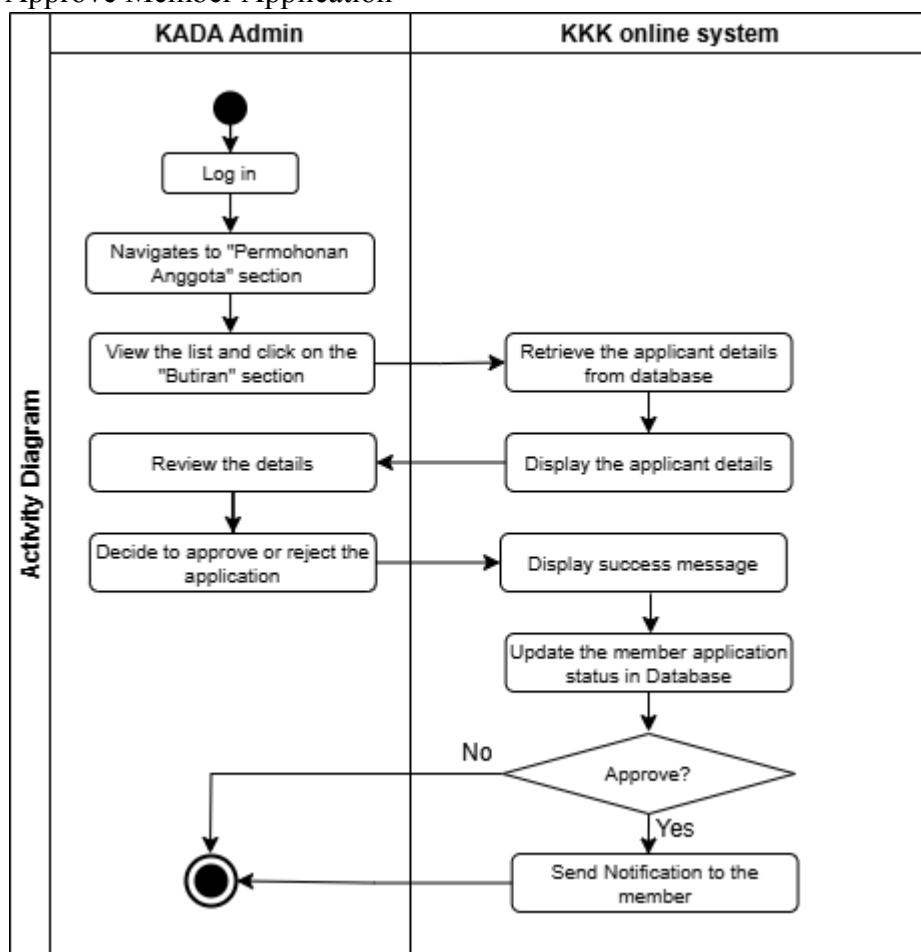


View Loan Application Status

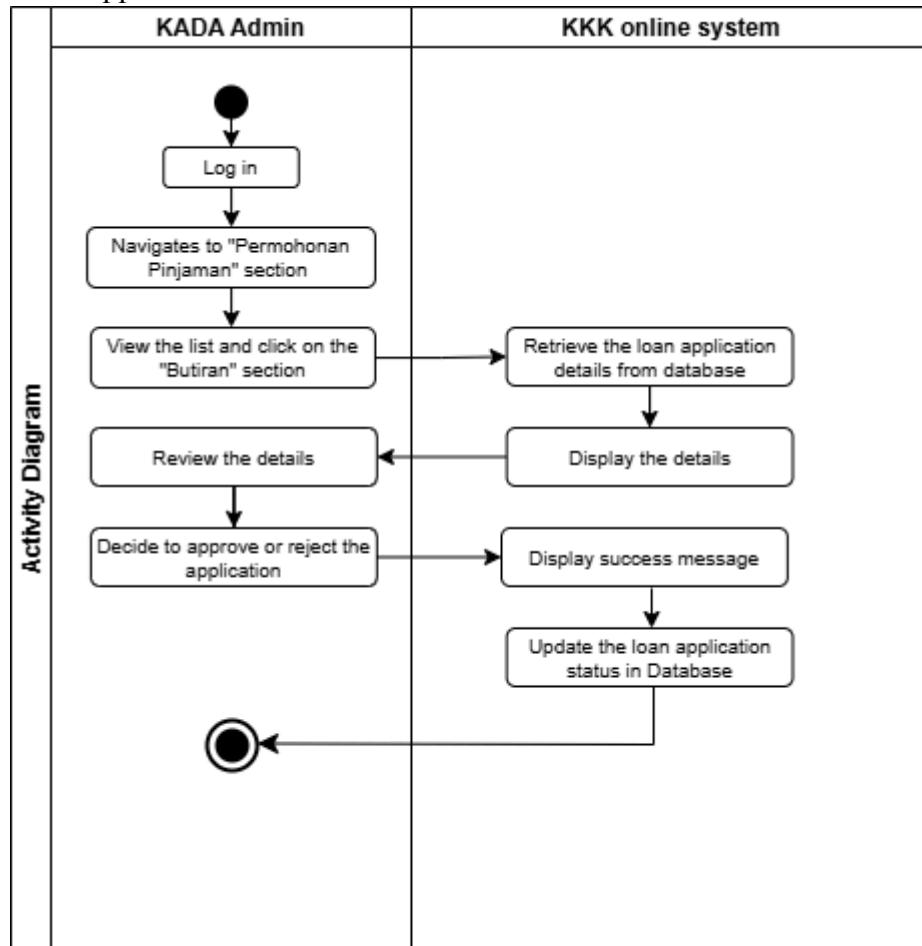


Admin

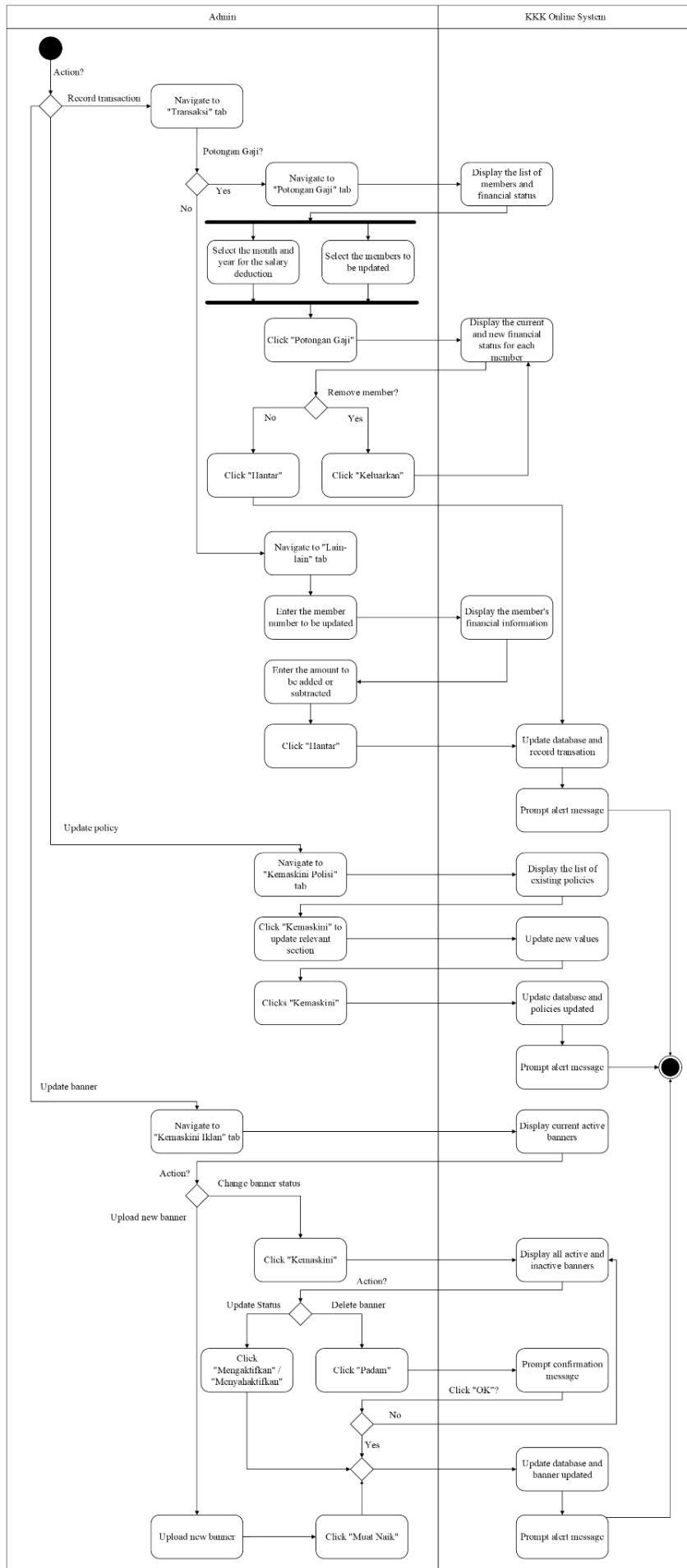
Approve Member Application



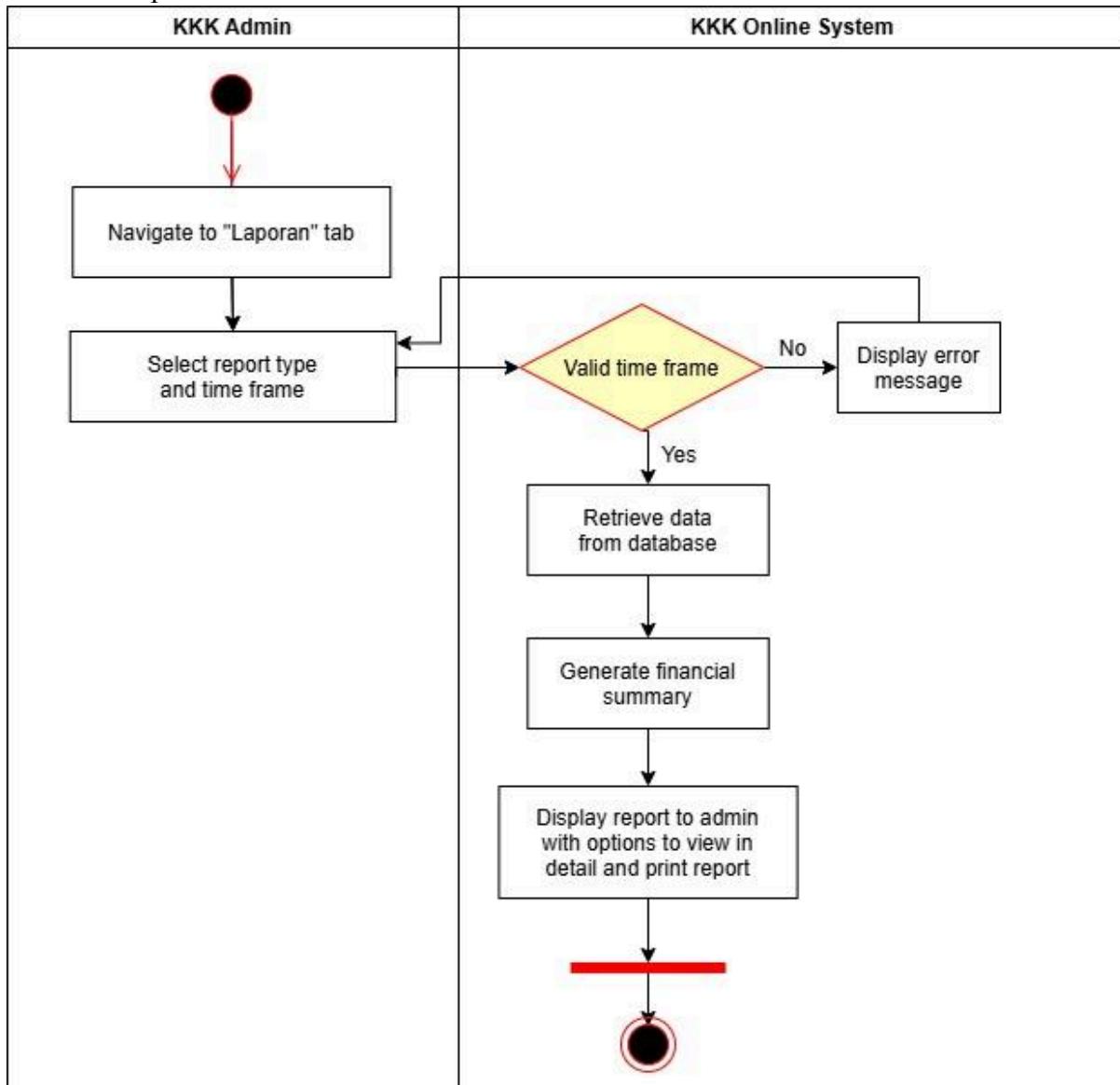
Loan Approval



Manage System



Generate Report



5.3 Meeting Log

Meeting Minutes	
Date	15 January 2025
Topic	P3 Report Content Ideas and Task Distribution
Attendees	Chua Jia Lin Goe Jie Ying Lam Yoke Yu Tan Yi Ya Teh Ru Qian

Action Items	
Parts	Person In Charge
1 Introduction	Lam Yoke Yu
2.1 Global ERD	Chua Jia Lin
2.2 Data Dictionary	Chua Jia Lin
3.1 Logical ERD	Goe Jie Ying
3.2 Data Dictionary	Goe Jie Ying
3.3 Relational Database Schema	Teh Ru Qian
3.4 Normalisation	Tan Yi Ya
4 Interface Design and SQL Implementation	All members will complete respective modules
5 Summary	Lam Yoke Yu
6.1 Use Case Diagram	Lam Yoke Yu
6.2 Activity Diagram	All members will complete respective modules
6.3 Meeting Log	Lam Yoke Yu